Der DATA BECKER Führer

CP/M

Alles auf einen Blick

ISBN 3-89011-400-8

Copyright © 1986 DATA BECKER GmbH
Merowingerstraße 30
4000 Düsseldorf

Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form (Druck, Fotokopie oder einem anderen Verfahren) ohne schriftliche Genehmigung der DATA BECKER GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.*

Wichtiger Hinweis:
Die in diesem Buch wiedergegebenen Schaltungen, Verfahren und Programme werden ohne Rücksicht auf die Patentlage mitgeteilt. Sie sind ausschließlich für Amateur- und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden.

Alle Schaltungen, technischen Angaben und Programme in diesem Buch wurden von dem Autoren mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. DATA BECKER sieht sich deshalb gezwungen, darauf hinzuweisen, daß weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernommen werden kann. Für die Mitteilung eventueller Fehler ist der Autor jederzeit dankbar.

Inhalt

Befehlsübersicht 9

CP/M-Führer 11

Residente Kommandos 13

Transiente Kommandos 21

Fehlermeldungen 87

BDOS und BIOS 119

Handhabung 127

Literaturverzeichnis 131

Index 133

Befehlsübersicht

Residente Kommandos 13
AMSDOS 14
DIR 15
DIRS(YS) 16
ERASE 17
RENAME 18
TYPE 19
USER 20

Transiente Kommandos 21
ASM 22
COPYSYS 23
DATE 24
DDT 25
DEVICE 26
DIR 28
DUMP 30
ED 31
ERASE 35
FORMAT 36
GENCOM 37
GET 38
HELP 40
HEXCOM 42
INITDIR 43
KEYFIG 44
LIB 45
LINK 46
LOAD 49
MAC 50
MOVCPM 51
PATCH 52
PIP 53
PUT 58
RENAME 60
RMAC 62
SAVE 63
SET 64
SETDEF 69
SETUP 71
SHOW 73
SID 74
SUBMIT 77
STAT 79
SYSGEN 83
TYPE 84
XREF 85

CP/M-Führer

Zu CP/M

CP/M steht für *C*ontrol *P*rogram for *M*icrocomputers und ist ein Kind der amerikanischen Firma Digital Research. CP/M heißt also nichts anderes als "Kontrollprogramm für Mikrocomputer".

Sowohl auf der Anwender- als auch auf der Programmiererseite sollte CP/M einiges revolutionieren. Wie häufig hat sich der Anwender geärgert, daß ein beliebig erworbenes Programm X nicht auf jedem Rechner läuft. Was liegt also näher, als einen Standard zu entwickeln - überall in der Welt und in allen Branchen ist man bemüht, sich auf solche Standards zu einigen.

Wo man einen Standard schafft, muß man auch eine Basis haben: CP/M läuft nur auf Rechnern mit der 8080 CPU. Aber auch die Rechner, die eine 8080-aufwärtskompatible CPU besitzen, sind CP/M-fähig, also Rechner mit der 8085 und speziell mit der Z-80. Ferner müssen CP/M-Programme in Maschinensprache kodiert werden. Später entwickelte man auch entsprechende Compiler für Hochsprachen wie etwa BASIC.

Wegen der Attraktivität des CP/M hat man auch für die 8086 (IBM-PC und Kompatible) das CP/M 86 geschaffen sowie für den 68000 das CP/M 68K (Atari und Einplatinencomputer). Verschiedene Namen zwar, von der Bedienung her aber vollkommen identisch. Einzige Ausnahme ist die Benutzung der Assembler und der Debugger-Utilities.

Die Handhabung dieses Führers

In diesem DATA-BECKER-Führer beschäftigen wir uns mit den beiden CP/M-Versionen CP/M 2.2 und CP/M 3.0 (auch CP/M-Plus genannt) sowie CP/M 86 und CP/M 68K, die an CP/M 3.0 angelehnt sind. Der eigentliche Unterschied zwischen den beiden Versionen 2.2 und 3.0 ist die Speichergröße, die von ihnen unterstützt bzw. vorausgesetzt wird. CP/M 2.2 kann man

auf Rechnern mit mindestens 64 KByte RAM "fahren", CP/M 3.0 hingegen benötigt wenigstens 128 KByte RAM. CP/M 68K und CP/M 86 haben meistens den für CP/M 3.0 erforderlichen Speicherplatz von 128 KByte zur Verfügung (bei IBM-PC und Atari) und sind deshalb mit den CP/M-3.0-Kommandos zu bedienen.

Die Kommandos des CP/M sind untergliedert in *transiente* und in *residente*. Residente Kommandos sind diejenigen, die nach dem Booten von CP/M stets zur Verfügung stehen. Anders die transienten; sie kann man nur dann benutzen, wenn sich die Systemdiskette im Laufwerk befindet.

Sofern notwendig, wird bei einem CP/M-Kommando erwähnt, ob es nur für eine der genannten CP/M-Versionen gültig ist. Ferner existieren noch eine Reihe *spezieller* CP/M-Kommandos, die rechnerspezifisch sind. Auch diese Kommandos werden speziell markiert, beispielsweise durch *C128* für ein Kommando, das auschließlich auf dem Commodore 128 existiert.

Die bei den CP/M-Kommandos angegebene Syntax gibt stets *alle* möglichen Eingabeformate wieder. Dabei haben einige Sonderzeichen folgende Bedeutungen:

{} Platzhalter werden in Klammern eingeschlossen, beispielsweise {Dateiname}. Beginnt der Platzhalter mit einem Komma, so ist die Eingabe optional: {,Option}

| Linke *oder* rechte Option: [PAGE | NOPAGE]

d Laufwerksbezeichnung (A bis P): DIR {d:}

Der *Index* enthält Stichworte sowie CP/M-Kommandos; beispielsweise erfahren Sie, wie Sie eine Diskette formatieren unter *FORMAT* sowie unter *Diskette formatieren*. Die CP/M-Kommandos selbst finden Sie zusätzlich in der Befehlsübersicht auf Seite 9.

Die residenten Kommandos

AMSDOS *(Nur CPC)*

Zurück zum AMSDOS-BASIC.

Syntax:

```
AMSDOS
```

Beschreibung:

So wie die Möglichkeit besteht, vom AMSDOS heraus durch das Kommando |CPM die Versionen CP/M 2.2 bzw. CP/M 3.0 aufzurufen, besteht auch die Möglichkeit, CP/M wieder zu verlassen und ins AMSDOS-BASIC zurückzukehren. Bedenken Sie, daß CP/M auch Speicher benötigt und daß Ihr BASIC-Programm oder Ihre Daten nach Aufruf des CP/Ms nicht unbedingt unversehrt sind.

Die Dateien von AMSDOS und CP/M sind weitgehend kompatibel. Sie können sich sogar eine Eigenschaft des CP/M zunutze machen: Ändern Sie durch das Kommando STAT oder SET das Attribut einer Datei auf Read-Only (Nur Lesen), so kann diese Datei auch von *AMSDOS* aus nicht mehr gelöscht oder überschrieben werden; Sie erhalten dann vielmehr eine entsprechende Fehlermeldung.

Dieses Attribut kann von AMSDOS höchstens durch entsprechende Maschinenroutinen, die direkt die CP/M-Routinen des AMSDOS-ROM nutzen (siehe auch Floppybuch zum CPC), wieder aufgehoben werden.

DIR

Residentes DIR: Anzeigen aller oder bestimmter Dateien.

Syntax:

```
DIR {d:}
DIR {d:}{Dateiname}
```

Beschreibung:

Hier handelt es sich um den residenten Teil des *DIR* (beim CP/M 2.2 auch der einzige), bei dem *keine* Optionen eingegeben und nicht mehrere Pfaddateien eingegeben werden können. *DIR* dient dem Ausgeben des Inhaltsverzeichnisses einer Diskette in gesamter Form oder nur partiell.

DIR existiert als residentes und als transientes Kommando. Das transiente *DIR* wird nur dann ausgeführt, wenn Optionen in eckigen Klammern folgen; siehe hierzu aber auch *DIR* als transientes Kommando. Stehen auch System-Dateien im Inhaltsverzeichnis, erscheint die Meldung:

```
SYSTEM FILE(S) EXIST
```

Bedienen Sie sich dann des DIRS-Kommandos (bei CP/M 2.2 STAT).

—» *DIR (transient), DIRS*

DIRSYS *(CP/M 3.0)*

Ausgabe aller Dateien, die das System-Attribut besitzen.

Syntax:

```
DIRSYS
DIRSYS {d:}
DIRSYS {d:}{Dateiname}
DIRS   {d:}{Dateiname}
```

Beschreibung:

DIRSYS ist ein residentes Kommando, deshalb arbeitet es schneller als das entsprechende *DIR*-Kommando mit System-Option (transientes *DIR* muß nicht nachgeladen werden). DIRSYS listet nur Dateien auf, die das System-Attribut besitzen. DIRSYS kann auch zu

```
DIRS
```

abgekürzt werden. Ferner erhalten Sie beim *DIRSYS*-Kommando noch Auskunft darüber, ob es neben den System-Dateien noch "normale" - CP/M nennt diese Dateien Nicht-System-Dateien - existieren. Existieren keine System-Dateien, so erscheint auf dem Monitor die Meldung:

```
No File
```

—» *DIR, SET, STAT, SHOW*

ERA(SE)

Löschen von Dateien.

Syntax:

```
ERASE {Dateiname}
ERASE {d:}{Dateiname}
```

Beschreibung:

Das transiente ERASE wird nur dann ausgeführt, wenn Sie eine Option [CONFIRM] oder kurz [C] angeben. Siehe hierzu auch *ERASE* im Teil *transiente Kommandos.*

Wollen Sie alle Dateien einer Diskette oder alle Dateien einer Dateikennung löschen, so ist es notwendig, daß Sie diese Löschoperation extra bestätigen, da man nur allzuschnell alle Dateien auf einer Diskette aus Versehen gelöscht hat. Dies sieht dann ungefähr so aus:

```
A>ERA B:*.*
ERASE B:*.* (Y/N)? Y

A>ERA A:*.TXT
ERASE *.TXT (Y/N)? N
```

Gibt es die von Ihnen angegebene Datei nicht auf der angewählten Diskette, so meldet ERASE dies mit:

```
No File
```

REN(AME)

Datei umbenennen.

Syntax:

```
RENAME
RENAME {Neuer Dateiname} = {Alter Dateiname}
```

Beschreibung:

Mit *RENAME* wird der Dateiname einer Datei geändert. Der Inhalt dieser Datei bleibt unverändert.

Auch *RENAME* liegt in transienter sowie in residenter Form vor. Benutzen Sie *RENAME* ohne Joker (wild cards), so wird die residente Version ausgeführt; Sie können lediglich eine Datei umbenennen. Bedienen Sie sich aber der Joker, so muß *RENAME.COM* auf einer Diskette vorliegen (CP/M 3.0); Sie können nun mehrere Dateien auf einmal umbenennen.

Sollte sich beim Umbenennen herausstellen, daß eine bestehende Datei überschrieben würde, so fragt RENAME nach, ob die alte Datei gelöscht werden soll. Beantworten Sie diese Frage mit Nein, so wird das RENAME-Kommando abgebrochen.

—» *RENAME (transient)*

TYPE

Anzeigen des Dateiinhaltes auf dem Bildschirm.

Syntax:

```
TYPE
TYPE {d:}{Dateiname {[PAGE|NO PAGE]}}
```

Beschreibung:

Neben dem transienten TYPE gibt es noch das residente TYPE. Das transiente TYPE wird nur dann geladen, wenn die Option angegeben oder kein Dateiname angegeben wird.

Wird kein Dateiname angegeben, so meldet sich TYPE mit:

```
Enter filename:
```

Sie können die Ausgabe auf den Bildschirm mit CTRL-S und CTRL-Q beliebig anhalten und fortführen. Haben Sie die Option *NOPAGE* angegeben, so wird kontinuierlich auf dem Bildschirm ausgegeben und ohne anzuhalten gescrollt. Anders bei Angabe der Option *PAGE*; hier wird, sobald der Bildschirm gefüllt wurde, die Ausgabe angehalten. Erst wenn Sie eine beliebige Taste betätigt haben, wird mit der Ausgabe fortgefahren.

Haben Sie einen Dateinamen angegeben, der nicht existiert, meldet TYPE:

```
No File
```

Um die Ausgabe der Datei auf den Drucker umzuleiten, müssen Sie, bevor Sie die Datei mittels TYPE {Dateiname} ausgeben lassen, den Protokollmodus auf den Drucker mit CTRL-P einschalten.

USER

Anzeige oder Ändern des aktuellen Benutzerbereichs.

Syntax:

```
USER
USER (n)
```

Beschreibung:

Es ist möglich, jede Diskette in bis zu 16 unterschiedliche, voneinander unabhängige sogenannte Benutzerbereiche zu unterteilen. Auf diese Weise können Sie sich beispielsweise einen Benutzerbereich für Texte anlegen, einen anderen für Programme etc. Einen Sonderstatus nehmen die Systemdateien aus dem Benutzerbereich Null ein, sie sind von allen Benutzerbereichen aus aufrufbar.

Der aktuelle Benutzerbereich (einer dieser 16) wird im CP/M-Prompt vor dem Dateinamen angezeigt (nur CP/M-PLUS!!), beispielsweise: *2B>*. Geben Sie bei *USER* den Benutzerbereich nicht an, so wird dieser vom Programm abgefragt. Mit dem Kommando *SHOW* können Sie sich die Benutzerbereiche anzeigen lassen, die Dateien enthalten.

```
USER

Enter User#:4
4A>
```

Die transienten Kommandos

ASM

Erstellt aus einer Quelldatei in 8080-Mnemocode eine Zieldatei im Hex-Format.

Syntax:

```
ASM {d:}{Dateiname}
ASM {d:}{Dateiname.{Op1}{Op2}{Op3}}
```

Beschreibung

Assembler dienen dazu, aus Mnemocode maschinenverständlichen und ausführbaren Code zu machen. Der CP/M-Assembler ASM erstellt zunächst eine sogenannte Hex-Datei, die später mit LOAD bearbeitet werden muß. Berücksichtigen Sie bitte, daß *ASM* nicht Z-80-Mnemocode sondern vielmehr 8080-Mnemocode assembliert. Der 8080 ist der Vorgänger des Z-80.

ASM legt die Hexdatei unter {Dateiname}.HEX an, sowie eine Protokolldatei unter {Dateiname}.PRN. Beachten Sie, daß Sie bei ASM *keine* Dateikennung angeben dürfen. Assemblieren Sie einmal die auf der Diskette befindliche Beispieldatei:

```
ASM DUMP
```

Optionen:

Option {Op1} für ASM-Datei: A-P
Option {Op2} für HEX-Datei: A-P | Z
Option {Op3} für PRN-Datei: A-P | X | Z

A-P ist die Laufwerksangabe für die Zieldatei.
X bedeutet, daß diese Datei nicht angelegt werden soll.
Z bedeutet, daß die Ausgabe auf den Bildschirm soll.

—» *LOAD, RMAC*

COPYSYS *(CP/M 3.0)*

COPYSYS kopiert die Systemspuren und wahlweise auch die Datei CPM3.SYS.

Syntax:

```
COPYSYS
```

Beschreibung:

Um von einer Diskette CP/M 3.0 zu starten, müssen beide CP/M-Teile vorhanden sein: Der erste Teil auf den Systemspuren, der zweite als Datei CPM3.SYS. Disketten, die nicht zum Booten verwendet werden, brauchen beides nicht.

Beim Commodore 128 ist die eigentliche Form des COPYSYS-Kommandos allerdings leicht abgeändert, Sie können ja das System durch einfaches Kopieren mittels PIP übertragen. Sollten Sie dennoch - etwa aus Gewohnheit - das Kommando COPYSYS aufrufen, so wird Ihnen auf dem Bildschirm angezeigt, daß es dieses Kommando praktisch nicht mehr gibt.

DATE *(CP/M 3.0)*

Anzeige von Datum und Zeit sowie deren Eingabe.

Syntax:

```
DATE
DATE CONTINUOUS
DATE SET
DATE MM/TT/JJ SS:MM:SS
```

Beschreibung:

Das Kommando DATE dient dazu, sich einserseits Datum und Zeit anzeigen zu lassen und andererseits Datum und Zeit einzustellen. Nach dem Booten des Systems werden die Werte für Zeit und Datum auf die Werte eingestellt, die bei der Installation der Systemdiskette vorlagen. Somit ist es notwendig, die Werte zu Beginn zu korrigieren.

Geben Sie nur DATE ein, so werden Ihnen Datum und Zeit einmal ausgegeben. Wollen Sie eine stehende Anzeige, so geben Sie DATE CONTINOUS oder kurz DATE C ein. Die Zeitanzeige bleibt dann so lange stehen, bis Sie eine Taste betätigen.

```
DATE C

Sat 07/26/86 17:03:33
```

Es gibt zwei Wege, Datum und Zeit einzugeben. Erste Möglichkeit: Sie geben DATE SET ein, dann fragt Sie CP/M getrennt nach Datum und Zeit. Zweite Möglichkeit: Sie geben Datum und Zeit direkt im Kommando an. Beachten Sie auf jeden Fall, daß das Datum im amerikanischen Format eingegeben werden muß (MM/TT/JJ: Monat/Tag/Jahr). In beiden Fällen müssen Sie eine Taste betätigen, um die Uhr exakt zu stellen.

DDT *(CP/M 2.2)*

DDT ist ein Debugger-Programm.

Syntax:

```
DDT
DDT {d:}{Dateiname}
```

Beschreibung:

Das Programm DDT ersetzt den CCP *(Console Command Processor)* und lädt dann die angegebene Datei in den Arbeitsspeicher. Mittels DDT kann die nächste Adresse nach der Endadresse eines Programmes (NEXT) und der Programmzählerinhalt (PC) abgefragt werden. DDT ist also ein Monitorprogramm mit einigen wenigen Möglichkeiten.

Wird bei *DDT* kein Dateiname eingegeben, so fragt *DDT* danach. Beachten Sie, daß Sie ausschließlich COM-Dateien einladen können. *DDT* ist der Vorläufer von SID.

—» *SID, LOAD, ASM*

DEVICE *(CP/M 3.0)*

Zeigt und verändert die Wege zu den Peripherie-Geräten.

Syntax:

```
DEVICE {NAMES | VALUES | physical Device | logical Device}
DEVICE logical device = NULL
DEVICE physical device {option}
DEVICE CONSOLE [PAGE | COLUMNS = {Spalten} | LINES = {zeilen}]
DEVICE logical device = physical device {option} {,physical
                                          device{option},...}
```

Beschreibung:

DEVICE dient dazu, die Ein-/Ausgabekanäle von CP/M anzuzeigen sowie diese zu verändern. Man ordnet immer logischen Kanälen physikalische Kanäle zu. Dabei ist es nicht vorgeschrieben, daß einem logischen Kanal lediglich ein physikalischer Kanal zugeordnet werden kann. Folgende logische Kanäle sind vorhanden:

```
AUX: (= AUXIN: und AUXOUT:)          CONOUT:
AUXILIARY: (= AUXIN: und AUXOUT:)    CONSOLE: (= CONIN: und CONOUT:)
AUXIN:                               KEYBOARD: (= CONIN:)
AUXOUT:                              LST:
CON: (= CONIN: und CONOUT:)          PRINTER (= LST:)
CONIN:
```

Die Namen der Ausgabeeinheiten sind von Hersteller zu Hersteller verschieden. Geben Sie *DEVICE* mit der Option *NAMES* ein, dann bekommen Sie eine Liste der möglichen Namen sowie die Übertragungsraten der einzelnen Geräte. Durch DEVICE VALUES erhalten Sie eine Liste der aktuellen Zuordnung:

```
Current Assignments:

CONIN:  = CRT
CONOUT: = CRT
AUXIN:  = Null Device
AUXOUT: = Null Device
LST:    = LPT
```

CON: steht für *Con*sole, also zusammenfassend Konsolein- und -ausgabe. Genau wie bei AUX: können Sie CON: in Ein- und Ausgabe trennen. LST: bedeutet *List* Device und repräsentiert den Drucker, wie man in obiger Liste aber auch recht gut sehen kann (Zuordnung mit LPT:).

In Verbindung mit der PAGE-Option können Sie feststellen, wie viele Zeichen pro Zeile und wie viele Zeilen auf dem Bildschirm für die Ausgabe zur Verfügung stehen. Ferner können Sie mit dieser Option auf die Bildschirmgröße Einfluß nehmen, indem Sie beispielsweise die Zeichenzahl pro Zeile verringern.

Beachten Sie, daß das Kommando DEVICE unter CP/M 2.0 durch das Kommando STAT DEV: ersetzt wird (richtiger müßte man sagen, daß man das Kommando STAT unter CP/M 2.2 gesplittet vorfindet, da es recht komplex ist).

Bei seriellen Geräten können Sie die Baudraten sowie den XON/XOFF-Status definieren. Beispiel für Laserdrucker:

```
DEVICE LPT [XON, 9600]
```

Mögliche Baudraten sind:

50, 75, 110, 134, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200

DIR *(CP/M 3.0)*

Transientes DIR: Anzeigen aller oder bestimmter Dateien mit Angabe von Optionen.

Syntax:

```
DIR {d:}
DIR {d:}{[Optionen]}
DIR {d:}{Dateiname}
DIR {d:}{Dateiname} {,Dateiname{,Dateiname...}}
DIR {d:}{Dateiname} {[Optionen]}
```

Beschreibung:

DIR gibt eine Liste aller oder bestimmter Dateinamen mit den dazugehörigen Informationen über diese Dateien auf dem Bildschirm aus.

DIR existiert als residentes und als transientes Kommando. Das transiente DIR ist erheblich leistungsstärker und wird nur dann ausgeführt, wenn Optionen in eckigen Klammern folgen oder mehr als ein Dateiname angegeben wird. Beachten Sie, daß lediglich die Dateien aus dem aktuellen Benutzerbereich berücksichtigt werden. Die Verwendung von Jokern (wild cards) im Dateinamen ist erlaubt. Wird kein Dateiname angegeben, so werden alle Dateien des aktuellen Benutzerbereiches ausgegeben.

Wird nicht die System-Option gewählt, so werden ausschließlich die Nicht-Systemdateien ausgegeben. Trotzdem wird aber überprüft, ob es unter dem angegebenen Dateinamen auch Systemdateien gibt. Ist dies der Fall, so erscheint die Meldung:

```
SYSTEM FILE(S) EXIST
```

Optionen:

Option	Funktion
ATT	Gibt frei definierbare Dateiattribute F1, F2, F3 und F4 aus.
DATE	Zeitgestempelte Dateien anzeigen.
DIR	Zeigt Nichtsystemdateien.
DRIVE=ALL	Zeigt Dateien aller Laufwerke.
DRIVE=A	Zeigt Dateien des Laufwerks A:.
DRIVE=(A,B)	Zeigt Dateien der angegebenen Laufwerke.
EXCLUDE	Zeigt alle außer der ausgewählten Datei.
FF	Seitenvorschub vor Ausgabe des Verzeichnisses.
FULL	Alle Attribute anzeigen.
LENGTH=n	Neuer Listenkopf nach n Zeilen.
MESSAGE	Zeigt Laufwerke und Benutzerbereiche fortlaufend.
NOPAGE	Beim Scrollen wird nicht angehalten.
NOSORT	Dateien nicht sortieren.
RO	Zeigt nur Read-Only-Dateien.
RW	Zeigt nur Read-Write-Dateien.
SIZE	Jeweilige Dateigröße ausgeben.
SYS	Zeigt nur Systemdateien.
USER=ALL	Alle Benutzerbereiche berücksichtigen.
USER=2	Dateien des Benutzerbereiches berücksichtigen.
USER=(5,7)	Dateien mehrerer Benutzerbereiche berücksichtigen.

Selbstverständlich können die verschiedenen Optionen auch kombiniert eingegeben werden, beispielsweise

```
DIR [DRIVE=B, FF]
```

—» *DIR (resident), DIRSYS, STAT, SHOW*

DUMP

Auslisten von Dateien in Hexadezimal und ASCII.

Syntax:

```
DUMP
DUMP {Dateiname}
```

Beschreibung:

DUMP ist besonders dann interessant, wenn man sich Dateien ausgeben lassen will, die nichtdruckbare Steuerzeichen enthalten. Textdateien werden gewöhnlich mittels TYPE auf den Bildschirm oder Drucker gebracht.

Bei der Ausgabe auf den Bildschirm mittels *DUMP* wird die Adresse ausgegeben, gefolgt von je 16 Hexadezimalwerten, denen wiederum die Ausgabe der ASCII-Zeichen folgt. Nicht druckbare Zeichen werden durch einen Punkt (.) dargestellt. Sie können die Ausgabe jederzeit mit CTRL-C anhalten. Wünschen Sie einen Ausdruck auf Drucker, so betätigen Sie vor dem DUMP-Kommando die Tasten CTRL und P.

```
A> DUMP TEXT.TXT

DUMP - Version 3.0
0000: 41 42 43 0D 0A 44 45 46 47 0D 0A 4E 44 45 51 ABC..DEFG.ODER
0100: 1B 1B 1B 00 00 00 00 00 00 00 00 00 00 00 00 ..............
```

—» *MAC, RMAC*

ED

Unter CP/M integrierter Zeileneditor.

Syntax:

```
ED {{d:}Eingabedatei {{d:}Ausgabedatei}}
```

Beschreibung:

Mittels *ED* können Sie eine Textdatei erstellen oder editieren (verändern). Da *ED* nicht sehr benutzerfreundlich ist, stellt sich das Arbeiten mit *ED* als sehr mühsam dar.

ED ist ein zeilenorientierter Texteditor, was nichts anderes bedeutet, als daß Sie jede Zeile einzeln ansprechen müssen und auch nur einzeln ändern können.

Wird beim ED-Aufruf eine Dateiname mit angegeben, so überprüft *ED*, ob diese Datei bereits existiert. Sollte dies nicht der Fall sein, so erscheint auf dem Bildschirm die Meldung

```
NEW FILE
```

Um die Datei dann abzuspeichern und *ED* zu verlassen, geben Sie das E-Kommando. Siehe hierzu aber auch die folgende Auflistung der ED-Kommandos.

ED-Kommandos:

nA Hängt n Zeilen vom Quellfile an den ED-Buffer an.

0A Anhängen an ED-Buffer vom Quellfile, bis Buffer halbvoll ist.

#A Anhängen aus Quellfile, bis Buffer voll ist.

B Bewegt CP *(Character Pointer)* zum Anfang des Buffers.

-B Bewegt CP zum Ende des Buffers.

nC Bewegt CP um *n* Zeichen forwärts im Buffer.

-nC Bewegt CP um *n* Zeichen rückwärts im Buffer.

nD Löscht *n* Zeichen ab CP.

-nD Löscht *n* Zeichen hinter CP.

E Abspeichern der neuen Datei und zurück zu CP/M.

F Suchen des Strings F*{string}*^Z.

H Abspeichern der neuen Datei und dann erneuter Aufruf von *ED* und der neuen Datei, um diese wieder zu editieren (Zweck: Buffer löschen).

I Einschalten des Einfüge-Modus. Betätigen Sie CTRL-Z oder *ESCAPE*, um Einfüge-Modus wieder zu verlassen.

I I{*string*}^Z Einfügen des String an CP. Ein großgeschriebenes I erzwingt, daß alle eingegebenen Zeichen in Großbuchstaben umgewandelt werden. Geben Sie das Kommando i als Kleinbuchstabe ein, so wird die Groß-/Kleinschreibung des Strings berücksichtigt.

nK Löscht *n* Zeilen ab CP.

-nK Löscht *n* Zeilen vor CP.

nL Bewegt CP um *n* Zeilen.

-nL Bewegt CP um *n* Zeilen nach vorne.

0L Bewegt CP an den Anfang des Buffers.

nM n{Kommando}M: Führt {Kommando} n-mal aus.

n Bewegt CP um *n* Zeilen und zeigt diese Zeile an.

-n Bewegt CP um *n* Zeilen nach vorne und zeigt diese Zeile an.

n: Bewegt CP zur angegebenen Zeile.

N N{*string*}^Z: Erweitert Suchstring.

O Rückkehr zum Originalfile.

nP Bewegt CP um *n* Zeilen und zeigt diese Zeilen an.

-nP Bewegt CP um *n* Zeilen nach vorne und zeigt diese Zeilen an.

Q Beenden von ED, ohne die neue Datei abzuspeichern.

R R^Z: Liest Datei X???????.LIB in den ED-Buffer.

R R{*Dateiname*}^Z: Liest Datei in den Buffer.

S S{*Delete_string*}^Z{*Insert_string*}^Z: Ersetzt {*Delete_string*} durch {*Insert_string*}.

nT Zeigt *n* Zeilen nach unten bzw. bei -nT *n* Zeilen nach oben an.

U Übersetzen in Großbuchstaben.

-U Übersetzen in Kleinbuchstaben.

V Zeilennumerierung einschalten.

-V Zeilennumerierung ausschalten.

0V Verbleibenden Buffer anzeigen.

nW Schreibe *n* Zeilen zur Update-Datei.

nX nX^Z: Schreiben oder Anhängen von *n* Zeilen an die Datei X???????.LIB.

0X 0X^Z: Löschen der Datei X???????.LIB.

0X 0X{*Dateiname*}^Z: Löschen der Datei *dateiname*.

nZ Warte *n* Sekunden .

ERA(SE) *(CP/M 3.0)*

Löscht eine, mehrere oder alle Dateien von der Diskette.

Syntax:

```
ERASE {d:}{Dateiname}{[CONFIRM]}
```

Beschreibung:

Anders als beim residenten ERASE wird beim transienten ERASE, das durch Angabe der Option [CONFIRM] oder auch kurz [C] erzwungen wird, eine Auswahl der Dateien auf dem Bildschirm angezeigt. Der Benutzer kann dann jede einzelne angezeigt Datei zum Löschen markieren.

Wollen Sie alle Dateien einer Diskette oder alle Dateien einer Dateikennung löschen, so ist es notwendig, daß Sie diese Löschoperation extra bestätigen, da man sich nur allzuschnell alle Dateien auf einer Diskette aus Versehen gelöscht hat. Dies sieht dann ungefähr so aus:

```
A>ERA B:*.*
ERASE B:*.* (Y/N)? Y

A>ERA A:*.TXT
ERASE *.TXT (Y/N)? N
```

Gibt es die von Ihnen angegebene Datei nicht auf der angewählten Diskette, so meldet ERASE dies mit:

```
No File
```

FORMAT

Formatiert ein Diskette

Syntax:

```
FORMAT
```

Beschreibung:

Jede Diskette, die man für die Speicherung von Daten verwenden will, muß formatiert werden. Beim Formatieren einer Diskette werden alle auf der Diskette befindlichen Daten unwiderruflich gelöscht. FORMAT ist kein Standard-CP/M-Programm, weshalb es sein kann, daß es auf Ihrer Systemdiskette nicht vorhanden ist. Suchen Sie dann nach dem Programm COPY.COM und sehen Sie nach, ob Sie damit eine Diskette formatieren können.

Sie können solange Disketten formatieren, bis Sie *FORMAT* mit CTRL-C abbrechen oder auf die Frage, ob eine weitere Diskette formatiert werden soll, mit Nein antworten.

Anmerkung:

Beim Commodore 128 sind verschiedene Disketten-Formate möglich. Entsprechend bekommen Sie beim Formatieren auch angeboten, ob Sie einseitig oder zweiseitig formatieren wollen etc. Sie können beim FORMAT-Kommando ausschließlich Disketten im A-Laufwerk formatieren.

Beim Schneider-CP/M befindet sich das Formatierungsprogramm unter dem Hilfsprogramm DISCKIT3.

GENCOM

Erstellung einer speziellen Version des CP/M 3.0.

Syntax:

```
GENCOM {COM-Datei}{RSX-Datei}{[LOADER|NULL|SCB=(offset,wert)]}
```

Beschreibung:

GENCOM erstellt spezielle COM-Dateien, die mit RSX-Dateien verknüpft sind. RSX-Dateien sind residente Systemerweiterungen (*R*esident *S*ystem E*x*tensions). Mittls *GENCOM* wird eine spezieller Header für diese Dateien angelegt, die die Datei als RSX-Datei deklariert. Mittels *GENCOM* können Sie aber ebensogut eine zur RSX-Datei gemachte Datei wieder zu einer normalen COM-Datei machen. Folgende drei Optionen gibt es für GENCOM:

LOADER: Diese Option gibt an, daß keine weiteren Erweiterungen an die COM-Dateien gebunden sind.

NULL: Diese Option gibt an, daß ausschließlich RSX-Dateien definiert sind. CP/M legt hier eine sog. DUMMY-COM-Datei an.

SCB: System Control Block. Hier wird der SCB anhand der übergebenen Hexwerte (Offset, Value) gesetzt.

GET *(CP/M 3.0)*

Eingabedatei wird definiert.

Syntax:

```
GET CONSOLE INPUT FROM FILE {Dateiname}
GET CONSOLE INPUT FROM CONSOLE
GET CONSOLE INPUT FROM FILE {Dateiname} [SYSTEM]
GET FILE {Dateiname} [{NO ECHO|ECHO | SYSTEM]
```

Beschreibung:

Normalerweise kommen die Daten für die Eingabe über die Tastatur (Console). Diesen Umstand kann man korrigieren, indem man sich des GET-Kommandos bedient. Sicherlich reizvoll ist dieses Kommando beispielsweise bei SUBMIT-Dateien, um ganze Eingabeketten vorprogrammieren zu können.

Die Eingabedatei kann dann Systemkommandos beinhalten oder Eingaben für andere Programme, beispielsweise für ED. Benutzen Sie die SYSTEM-Option, so werden ausschließlich Systemkommandos akzeptiert, die anderen Zeilen werden überlesen.

Die Eingabe wird so lange von einer Datei geholt, bis entweder das Programm, das sich des *GET*-Kommandos bedient, zu Ende ist, oder das Ende der Eingabedatei erreicht wurde. Ist letzteres der Fall, so werden die weiteren geforderten Eingaben wieder von der Tastatur verlangt.

Bei Angabe der SYSTEM-Option wird direkt mit der Eingabe von der Eingabedatei fortgefahren - anders ohne diese Option: Sie können zunächst noch ein Systemkommando eingeben, erst dann werden alle Eingaben über diese Datei abgewickelt. Auf diese Weise können Sie noch ein Benutzerprogramm aufrufen.

Die Eingabe wird wieder von Tastatur gefordert, sobald das Ende der Datei erreicht oder das Kommando

```
GET CONSOLE INPUT FROM CONSOLE
```

ausgeführt worden ist.

Optionen:

ECHO	Die von der Datei geholten Eingaben werden auf dem Bildschirm angezeigt.
NO ECHO	Die von der Datei geholten Eingaben werden nicht auf dem Bildschirm angezeigt.
SYSTEM	Alle Systemeingaben werden aus der angegeben Datei geholt. Dieser Vorgang wird abgebrochen, wenn die Datei auf EOF stößt oder ein anderes GET-Kommando gefunden wird.

—» *PUT*

HELP *(CP/M 3.0)*

Anzeigen von Hilfeseiten - menüorientiert.

Syntax:

```
HELP
HELP {Stichwort}
HELP {Stichwort {Unterbegriff1 Unterbegriff2...Unterbegriff8}}
HELP {Stichwort} [NO PAGE | LIST]
HELP .{Unterbegriff}
HELP [CREATE | EXTRACT]
```

Beschreibung:

HELP ist ein Hilfsprogramm zur Unterstützung des CP/M-Benutzers. Wird *HELP* ohne Stichwort aufgerufen, so wird Ihnen eine Liste aller möglichen Stichworte sowie deren Unterbegriffe angezeigt. Wird *HELP* mit einem Stichwort aufgerufen, so werden Ihnen direkt die entsprechenden Informationen und die eventuell vorhandenen Unterbegriffe angezeigt.

Das Kommando HELP ist besonders für Anfänger sehr hilfreich, da man das Handbuch zu CP/M nicht immer parat haben muß. Auch in anderen bekannten Software-Paketen wurde diesem Beispiel von Digital Research nachgeeifert - wieviele Programme haben mittlerweile die sogenannten HELP-Screens integriert, die einem die Arbeit mit diesen Programmpaketen erleichtern sollen.

Die Stichwörter und die Unterbegriffe können auf die jeweils ersten zwei Buchstaben verkürzt eingegeben werden. Wird das Bildschirmende erreicht, so stoppt die Anzeige, und es wird durch einen beliebigen Tastendruck mit der Ausgabe fortgefahren. Einen Ausdruck der (zumeist in englischer Sprache verfaßten) Hilfsseiten erreichen Sie, indem Sie vor Aufruf des Begriffes CTRL-P betätigen.

Hat *HELP* Ihnen die gewünschte Hilfeseite gezeigt, meldet sich das Programm mit dem typischen Prompt "HELP>". Wollen Sie einen weiteren Unterbegriff sehen, so können Sie dies durch Voranstellen eines Punktes vor den Unterbegriff erreichen (ohne Punkt würde HELP annehmen, es handele sich nicht um einen Unterbegriff, sondern um einen Hauptbegriff). Um *HELP* zu verlassen, betätigen Sie ohne Eingabe einfach *<RETURN>*.

Optionen:

NO PAGE Anders als bei der Option LIST (das Gegenteil dieser Option) wird bei NOPAGE nicht bei Erreichen des unteren Bildschirmrandes die Ausgabe angehalten.

EXTRACT Die Datei HELP.HLP wird kopiert unter dem Namen HELP.DAT. Diese Datei können Sie mit einer Textverarbeitung nun bearbeiten oder ergänzen.

CREATE Durch diese Option werden eventuell angefertigte Änderungen in der Hilfsdatei HELP.DAT berücksichtigt. Es wird eine neue Datei HELP.HLP erstellt (mit Textänderungen, neuen Suchbegriffen oder Unterbegriffen).

HEXCOM *(CP/M 3.0, beim C128 Additional Utilities)*

Erstellt eine COM-Datei.

Syntax:

```
HEXCOM {Dateiname}
```

Beschreibung:

HEXCOM wandelt die angegebene Datei {Dateiname}, die im Hex-Format vorliegen muß, in eine ausführbare COM-Datei um. Die Dateikennung *HEX* ist obligatorisch und darf nicht mit eingegeben werden. *HEXCOM* ist der Nachfolger des CP/M-2.2-Kommandos LOAD.

Beachten Sie, daß HEXCOM nicht HEX-Dateien in COM-Dateien umwandelt, die von *RMAC*, also den relokatiblen Code erstellenden Assembler, erzeugt worden sind. Hierfür müssen Sie sich des Programmes LINK bedienen.

—» *LOAD, LINK, MAC, RMAC*

INITDIR *(CP/M 3.0)*

Diskette soll mit Zeit- und Datumsstempeln versehen werden.

Syntax:

```
INITDIR <d:>
```

Beschreibung:

Will man die Dateien auf einer Diskette mit Zeit- und Datumsstempeln versehen, beispielsweise um den letzten Zugriff zu prüfen oder das automatische Updaten zu ermöglichen, so ist es notwendig, die Diskette entsprechend vorzubereiten.

INITDIR organisiert das Inhaltsverzeichnis einer Diskette neu. Es wird für jede Datei mehr Platz vorgesehen, um die zusätzlich anfallenden Daten speichern zu können. Dadurch wird allerdings die Anzahl der möglichen Dateien pro Diskette halbiert. Am besten ist es, Sie bearbeiten eine Diskette direkt nach dem Formatieren mit *INITDIR*. Wollen Sie eine bereits mit Daten versehene Diskette umformatieren, so empfiehlt sich eine Sicherheitskopie, da bei frühzeitigem Abbruch des Programmes (beispielsweise durch CTRL-C) alle Daten auf der Diskette verloren sind.

Ist für das Umformatieren nicht mehr genügend Speicherplatz auf der Diskette, so meldet dies *INITDIR* und bricht den Arbeitsvorgang ab. Hat *INITDIR* seine Arbeit erfolgreich abgeschlossen, so müssen Sie die Diskette umgehend mit *SET* bearbeiten und die Datumsoption setzen.

—» *SET, DIR, PIP*

KEYFIG *(C128)*

Tastaturbelegung ändern.

Eingabeformat

```
KEYFIG
```

Beschreibung:

KEYFIG ist vollkommen menüorientiert. Man kann durch die <CTRL>-<RSHFT>-Tastenkombinationen zumeist dasselbe ohne KEYFIG erreichen, allerdings ist diese Anwendung etwas komplizierter und unübersichtlicher. Für den geübten Anwender erspart diese Methode allerdings auch Zeit.

KEYFIG hält dagegen HELP-Texte (Hilfen) für den Anwender bereit. Mittels *KEYFIG* sind Sie Herr über die Tastatur: Sie können jeder Taste, ob mit <Shift> oder mit <CTRL>, beliebige ASCII-Werte zuordnen. Ferner haben Sie die Möglichkeit, die Bildschirmfarben zu definieren und den Tasten Spezialfunktionen zuzuordnen (wie beispielsweise das Rebooten des Systems o.ä.). Als Krönung kann man dann den Tasten noch ganze Zeichenketten zuordnen, so wie es beispielsweise bei allen Funktionstasten und der HELP-Taste der Fall ist. Alles in allem: Eine sehr nützliche Sache. Allerdings müssen Sie dieses Kommando mit dem Vorbehalt benutzen, daß Sie ein ähnliches Programm auf einem anderen CP/M-Rechner nicht unbedingt erwarten dürfen.

LIB *(CP/M 3.0, beim C128 Additional Utilities)*

Anzeigen und verändern Bibliotheksinhalt.

Syntax:

```
LIB {Dateiname} {[I|M|P|D]}
LIB {Dateiname} {[I|M|P|D]}={Dateiname}{,Dateiname...}
```

Beschreibung:

Eine Bibliotheksdatei enthält mehrere Module (Unterprogramme und Funktionen), eine Bibliothek wiederum mehrere dieser Dateien. Mittels *LIB* können Sie Bibliotheken erstellen, verändern, ersetzen, anhängen, löschen und vieles mehr.

Die mittels *LIB* verwalteten REL-Dateien enthalten Objektcode im MicroSoft-REL-Format, wie sie beispielsweise vom CP/M-Assembler RMAC erstellt werden.

Die Optionen:

I Die Index-Option erstellt eine indizierte Bibliotheksdatei vom Typ IRL. Link-80 kann bei indizierten Bibliotheksdateien schneller suchen als bei nicht-indizierten.

M Die Modul-Option zeigt Modulnamen an.

P Die Publics-Option zeigt Modulnamen und die sog. Public-Variablennamen an.

D Die DUMP-Option zeigt den Dateiinhalt als Hexdump an.

—» *LINK, RMAC*

LINK *(CP/M 3.0, beim C128 Additional Utilities)*

Zusammenführen von Objektdateien aus der Bibliothek.

Syntax:

```
LINK {d:}{Dateiname{[O]}}={Dateiname{[O]}{...}
```

Beschreibung:

LINK bindet aus verschiebbarem Code, wie er beispielsweise von *RMAC* erzeugt wird, bestehende Dateien zu einer COM-Datei zusammen. Innerhalb dieser Dateien können Zugriffe auf externe Werte vorgenommen werden, die in anderen Dateien definiert werden. *LINK* hat die Aufgabe, diese Namen in der Bibliothek zu finden und zu ersetzen.

Auf diese Weise kann man beispielsweise sehr große Programme programmieren, indem man diese in mehrere kleine Teile aufsplittet. Voraussetzung ist allerdings, daß beim Augenblick des Compilierens noch keine feste Speicheradresse zugeordnet worden ist, da diese Zuordnung vom Linker erfolgt. Der Code muß also verschiebbar (relokatibel) sein.

Optionen:

A	Zusätzlicher Speicher wird angefordert. Der Buffer wird verkleinert und temporäre Daten auf Diskette gesichert.
B	BIOS wird eingelinkt. Datensegment liegt an Seitengrenze (Banking); die Länge des Codes wird im Dateiheader verzeichnet; Defaultoption für SPR-Dateien.
Dhhhh	Startadresse für Datenbereich in Hexadezimal
Gn	Setzt die Startadresse des Programmes auf das Label n
Lhhhh	Ändert die Ladeadresse der COM-Datei von 0100H auf hhhh
Mhhhh	Angabe der notwendigen freien Speichergröße
NL	Keine Ausgabe der Symboltabelle auf Bildschirm
NR	Kein Anlegen einer Symboldatei
OC	COM-Datei erstellen (Default)
OR	RSP-Datei erstellen (Resident System Process)
OS	SPR-Datei erstellen (System Page Relocatable)
Phhhh	Programm-Startadresse definieren; die defaultmäßige Startadresse 0100H wird geändert
Q	Symbole werden bei der Ausgabe mit einem führenden Fragezeichen versehen
S	Datei als Bibliothek suchen

$Cd Ausgabe der Konsolemeldungen auf:
X = Konsole,
Y = Drucker,
Z = Keine Ausgabe (Zero Output)

$Id Laufwerksangabe für Hauptdateien (A - P)

$Ld Laufwerksangabe für Bibliotheksdateien (A - P)

$Od Objektdatei auf:
Z = Keine Objektdatei oder Laufwerk
A - P, sonst Laufwerk, von wo auch *LINK* geladen wurde

$Sd Ausgabe Symboltabelle:
Y = Drucker,
Z = Keine Symboltabelle oder Laufwerk
A-P, sonst Laufwerk, von wo auch *LINK* geladen wurde

—» *MAC, LIB*

LOAD *(CP/M 2.2)*

Erstellt aus einer HEX-Datei eine COM-Datei.

Syntax:

```
LOAD {Dateiname}
```

Beschreibung:

LOAD macht aus der von *MAC* erstellten Hex-Datei eine ausführbare COM-Datei. Sie können mit *LOAD* keine von *RMAC* erstellten Hex-Dateien bearbeiten, da diese verschiebbaren Code beinhalten.

Beachten Sie, daß Sie bei der Eingabe des Dateinamens keine Dateikennung angeben dürfen; die Dateikennung *.HEX* ist obligatorisch, da der Assembler *MAC* ebenfalls automatisch Hex-Dateien mit eben dieser Dateikennung erstellt.

LOAD ist nach jedem Assemblierungsvorgang aufzurufen, da man mit dem Hex-Code eines Programmes alleine nichts anfangen kann.

—» *MAC, LINK*

MAC (*CP/M 3.0, beim C128 Additional Utilities)*

Ein Makroassembler für Programme in 8080-Mnemonic.

Syntax:

```
MAC {Dateiname}
MAC {Dateiname} ${Optionen}
```

Beschreibung:

MAC ist ein Makroassembler und assembliert einen in 8080-Mnemonic geschriebenen Quelltext. Dabei wird allerdings nicht sofort eine COM-Datei erstellt, sondern vielmehr eine Hex-Datei im INTEL-Format. Diese Hex-Datei mit der Dateikennung *HEX* muß dann später mit *HEXCOM* in eine COM-Datei umgewandelt werden.

Neben der Hex-Datei erstellt *MAC* noch eine Datei mit der Symboltabelle (Dateikennung *SYM*) sowie eine Protokolldatei, die das komplette Assemblerlisting beinhaltet (Dateikennung *PRN*). Werden während der Assemblierung Fehler gefunden, so werden diese auf dem Bildschirm angezeigt.

Optionen:

A	Quelldatei Laufwerk (A-O)
H	Zieldatei Laufwerk (A-O, Z)
L	Quellaufwerk für LIB-Dateien
P	Ziellaufwerk für PRN-File (A-O, X, P, Z)
S	Ziellaufwerk für SYM-Dateien (A-O, X, P, Z)

z.B. MAC DEMO $PA AB HB SX

—» *RMAC, LINK, LIB, HEXCOM*

MOVCPM *(CP/M 2.2)*

Verschieben des Systemes in Schritten von 256 Byte.

Eingabeform:

```
MOVCPM {Faktor}*
```

Beschreibung:

Der Befehl MOVCPM dient dazu, das System CP/M 2.2 in 256-Bytes-Einheiten zu verschieben, beispielsweise um Überschneidungen mit anderer Software zu vermeiden. Der einzugebende Faktor *<Faktor>* muß zwischen 64 und 179 liegen. Bei Angabe von 179 wird ein Standard-CP/M angelegt. Entsprechend wird beispielsweise bei *MOVCPM 178** ein um 256 Byte nach unten verschobenes CP/M erzeugt. Der *<Faktor>* gibt den für CP/M-Kommandos und Programme verfügbaren Speicherplatz in 256-Byte-Einheiten an.

Wird es aus irgend einem Grund notwendig, das System zu verschieben, beispielsweise weil sich BIOS/BDOS mit einem Anwenderprogramm überlagert, so kann man dies mittels *MOVCPM* erreichen. Das verschobene System kann man dann mit dem *SYSGEN*-Kommando verschieben.

—» *SYSGEN*

PATCH *(CP/M 3.0)*

Änderungen in CP/M 3.0 installieren.

Syntax:

```
PATCH {Dateiname}.{Typ} {n}
```

Beschreibung:

Mit *PATCH* können Sie eine Patchnummer *{n}* installieren oder sich die dazugehörigen Informationen ausgeben lassen. Die Patchnummer *{n}* muß zwischen 1 und 32 einschließlich liegen.

Lediglich Dateien mit den Endungen COM, PRL oder SPR können mit PATCH behandelt werden. Sollte die Dateikennung *{.Typ}* nicht angegeben sein, so nimmt *PATCH* die Endung *COM* an.

```
PATCH SHOW 2

Do you want to indicate that Patch #2
has been installed for SHOW.COM? Y

Patch Installed
```

PIP

Kopieren von Dateien.

Syntax:

```
PIP
PIP {d:}{Zieldatei}{[Optionen]}={Quelldatei}{[Optionen]}
PIP {Zieldatei}={Quelldatei1}{,Quelldatei2}{,Quelldatei3...}
```

Beschreibung:

PIP kopiert eine oder mehrere Dateien von einer Diskette auf eine andere oder von einem Benutzerbereich zu einem anderen. Mit *PIP* können Sie mehrere Dateien zu einer zusammenführen, das sogenannte *Updating* von veränderten Dateien erwirken, Dateien zum Drucker schicken und vieles mehr. Keiner, der mit CP/M arbeiten will, wird wohl um *PIP* herum kommen. *PIP* hat dementsprechend zahlreiche Optionen.

Kopieren Sie mit *PIP* Dateien, so werden auch alle zur Datei gehörenden Attribute mitkopiert, wie beispielsweise das Systemattribut oder der Read-Only-Status. Ist die zu kopierende Datei mit einem Paßwort belegt, so ist es erforderlich, daß Sie das Paßwort in der Kommandozeile mit eingeben. Stimmt das Paßwort nicht überein, so wird die Datei nicht kopiert und Ihnen dies auf dem Bildschirm gemeldet.

Wird *PIP* ohne Dateinamen eingegeben, so meldet sich *PIP* mit seinem Prompt *. Sie können dann beliebig viele Dateien kopieren, bis Sie eine Leereingabe machen, dann wird *PIP* verlassen.

Die Optionen in eckigen Klammern müssen direkt nach dem Dateinamen stehen, also ohne Leerzeichen. Die Verwendung von Jokern ist erlaubt.

Optionen:

A *Archiv-Funktion;* es werden nur die Dateien kopiert, bei denen das Archiv-Flag gesetzt ist, d.h. die seit dem letzten Backup erstellt oder verändert worden sind.

C *Confirm;* bei Jokerverwendung bedeutet dies, daß Sie jede zu kopierende Datei zunächst bestätigen müssen.

Dn *PIP* löscht alle Zeichen nach der n-ten Spalte. Dies ist beispielsweise dann sinnvoll, wenn man eine Datei vorliegen hat mit Zeilen, die mehr Zeichen enthalten, als eine Textverarbeitung einlesen kann.

E *Echo*; bei dieser Option wird der Inhalt der gerade kopierten Datei auf dem Bildschirm angezeigt.

F *Filter Form-Feeds;* um das Ausgabeformat auf Drucker nicht zu stören, das Sie ja mit der P-Option festlegen können, empfiehlt sich das Eliminieren von Form-Feeds aus der zu druckenden Datei.

Gn Hole oder bringe die Datei von oder zu Benutzerbereich n. Steht diese Option hinter der Zieldatei, so transportiert *PIP* diese Datei in den angegebenen Benutzerbereich. n muß zwischen 0 und 15 liegen.

H *Hex data transfer;* PIP überprüft beim Kopieren die Datei(en), ob diese das INTEL-Hex-Format einhalten. Tauchen Fehler auf, so werden diese Fehler auf dem Bildschirm angezeigt.

I *Ignore :00 records;* das End-Of-File-Zeichen wird beim Kopieren von Hex-Dateien ignoriert. Die I-Option setzt automatisch die H-Option.

L Diese Option ersetzt Großbuchstaben der Quelldatei in Kleinbuchstaben der Zieldatei. Sie sollten diese Option nur beim Kopieren von Textdateien verwenden.

N Diese Option numeriert die kopierten Zeilen durch, beginnend mit 1. Die Zeilennummern werden der Zieldatei hinzugefügt. Die Numerierung erfolgt in Einerschritten. Der Zeilennummer folgt ein Doppelpunkt und ein Leerzeichen. Die Zahlen sind sechsstellig und führende Nullen werden durch Leerzeichen ersetzt.

N2 Führende Nullen werden auch als Nullen dargestellt. Nach der Zeilennummer wird ein TAB eingefügt. Zusammen mit der T-Option wird dieser Tabulator erweitert.

O *Object file transfer;* wird zum Kopieren von Nicht-Textdateien (zumeist Maschinencode) verwendet. PIP ignoriert hier CTRL-Z sowie das Dateiendezeichen und kopiert diese nicht mit. Diese Option verwendet man, will man mehrere Objektdateien miteinander kombinieren.

Pn *Set page lenth;* die Seitenlänge wird mit n Zeilen definiert. Nach jeweils n Zeilen wird ein Line-Feed in die Zieldatei geschrieben sowie in die erste Zeile, sollte die Quelldatei durch diese Option verändert werden. Ist n=1 oder nicht angegeben, so wird nach jeweils 60 Zeilen ein Seitenvorschub eingefügt. Sie sollten diese Option in Verbindung mit der F-

Option verwenden, um eventuell vorhandene Seitenvorschübe aus der Quelldatei nicht zu berücksichtigen.

Qs^Z *Quit copying;* mit dieser Option geben Sie an, daß nicht die gesamte Datei kopiert werden soll, sondern daß nach dem Kopieren des Strings s das Kopieren abgebrochen wird. In Verbindung mit der S-Option können Sie auf diese Weise einen definierten Ausschnitt aus der Quelldatei kopieren. Der String muß mit einem CTRL-Z abgeschlossen sein.

R *Read SYS files;* normalerweise ignoriert *PIP* Dateien mit dem SYS-Attribut. Geben Sie diese Option an, so kopiert *PIP* auch diese Dateien mit deren Attributen.

Ss^Z *Start copying;* mit dieser Option können Sie angeben, daß Sie nicht die gesamte Quelldatei kopieren wollen, sondern vielmehr erst ab der Zeichenkette s. In Verbindung mit der Q-Option können Sie einen beliebigen Ausschnitt aus einer Datei herausholen und kopieren. Der String s muß mit einem CTRL-Z abgeschlossen sein.

Tn *Expand tabs;* geben Sie diese Option an, so ersetzt PIP alle Tabulatorzeichen (CTRL-I) durch die entsprechende Anzahl an Leerzeichen bis zum nächsten Tabulatorstopp teilbar durch n.

U Übersetzt Großbuchstaben der Quelldatei beim Kopieren in Kleinbuchstaben für die Zieldatei.

V *Verify; PIP* überprüft nach dem Kopieren der Datei bzw. der Dateien, ob alles korrekt kopiert worden ist. Werden Differenzen gefunden, so erscheint eine Fehlermeldung auf dem Bildschirm.

W Diese Option gibt an, daß Dateien mit dem Read-Only-Status überschrieben werden sollen. Normalerweise fragt PIP nach, ob eine bestehende schreibgeschützte Datei überschrieben werden darf. Geben Sie diese Option an, entfällt die Frage.

Z *Zero parity bit;* geben Sie diese Option an, so wird beim Kopieren der Quelldatei das 8. Bit von jedem Zeichen auf Null gesetzt. Beispielsweise ist diese Option beim Kopieren von WordStar-Dateien recht nützlich.

PUT

Umleiten der Bildschirmausgabe.

Syntax:

```
PUT CONSOLE {OUTPUT TO} FILE {Dateiname} {[OPTION]}
PUT PRINTER {OUTPUT TO} FILE {Dateiname} {[OPTION]}
PUT CONSOLE {OUTPUT TO} CONSOLE
PUT PRINTER {OUTPUT TO} PRINTER
```

Beschreibung:

Mittels des *PUT*-Kommandos können Sie es ermöglichen, daß die Daten, die zum Drucker oder zum Bildschirm gesandt werden, zusätzlich in einer Datei protokolliert werden. Dies ermöglicht beispielsweise spätere Fehleranalyse.

Ähnlich wie beim *GET*-Kommando haben Sie auch bei *PUT* die Möglichkeit, nach dem *PUT*-Kommando noch ein Systemkommando einzugeben oder ein Programm aufzurufen. Erst dann gehen die Ausgaben in die angegebene Datei, es sei denn, Sie geben die System-Option, dann gehen die Ausgaben *direkt* nach dem Kommando in die angegebene Datei.

Die Ausgabe auf Datei wird abgebrochen, wenn das entsprechend aufgerufene Systemprogramm beendet ist oder Sie *PUT CONSOLE* oder *PUT PRINTER* geben.

Syntax der Optionen:

```
[ECHO|NO ECHO | FILTER|NO FILTER | SYSTEM]
```

Optionen:

ECHO Diese Option gibt an, daß die in die Datei geschriebenen Ausgabezeichen ebenfalls auch auf dem Bildschirm angezeigt werden. Diese Option ist defaultmäßig beim PUT-Kommando eingeschaltet.

NO ECHO Diese Option gibt entsprechend an, daß die auszugebenden Zeichen nicht auf dem Bildschirm angezeigt, sondern lediglich in die Ausgabedatei geschrieben werden.

FILTER Diese Option gibt an, daß die auszugebenden Zeichen darauf geprüft werden, ob sie Steuerzeichen sind. Steuerzeichen werden dann in druckbare Zeichen umgewandelt, beispielsweise wird aus *ESCAPE* die Zeichenkette ^[.

NO FILTER Steuerzeichen werden entgegen der FILTER-Option nicht herausgefiltert. Dies ist defaultmäßig eingestellt.

SYSTEM Bei dieser Option werden sofort alle Ausgaben in die angegebene Ausgabedatei geschrieben, man kann also kein Kommando mehr eingeben. Die Ausgabe wird erst dann abgebrochen, wenn man dies mittels einem PUT-CONSOLE-Kommando verlangt.

—» *GET*

REN(AME) *(CP/M 3.0)*

Eine Datei umbenennen.

Syntax:

```
RENAME
RENAME {Neue Datei} = {Alte Datei}
```

Beschreibung:

Mit *RENAME* wird der Dateiname einer Datei geändert. Der Inhalt dieser Datei bleibt unverändert.

Auch *RENAME* liegt in transienter sowie in residenter Form vor. Benutzen Sie *RENAME* ohne Joker (wild cards), so wird die residente Version ausgeführt; Sie können lediglich eine Datei umbenennen. Bedienen Sie sich aber der Joker, so muß *RENAME.COM* auf einer Diskette vorliegen (CP/M 3.0); Sie können nun mehrere Dateien auf einmal umbenennen.

Sollte sich beim Umbenennen herausstellen, daß eine bestehende Datei überschrieben würde, so fragt *RENAME* nach, ob die alte Datei gelöscht werden soll. Beantworten Sie diese Frage mit Nein, so wird das *RENAME*-Kommando abgebrochen.

Sollte sich beim Umbenennen herausstellen, daß eine bestehende Datei überschrieben würde, so fragt *RENAME* nach, ob die alte Datei gelöscht werden soll. Beantworten Sie diese Frage mit Nein, so wird das *RENAME*-Kommando abgebrochen.

Ist die zu ersetzende Datei nicht vorhanden, so meldet CP/M sich mit

```
No File
```

Bei Existenz der umzubenennenden Datei in den neuen Namen meldet sich CP/M mit

```
Not renamed: filename.typ already exists,
delete(Y/N)?
```

Sie können nun angeben, ob die Datei durch die neue Datei überschrieben werden soll.

Das *RENAME* des CP/M 3.0 ist transient und die Ergänzung zum residenten *RENAME*. Das transiente *RENAME* wird nur dann ausgeführt, wenn im Dateinamen ein Joker vorkommt oder gar kein Dateiname angegeben wird.

```
REN *.TXT=*.BAK
REN W*.*=V*.*
```

Geben Sie *RENAME* ohne Dateinamen ein, so fragt RENAME nach den Dateinamen:

```
A>RENAME

Enter New Name:ALPHA.PRN
Enter Old Name:BETA.LST
BETA    .PRN=ALPHA   .LST
A>
```

Es reicht übrigens, wenn Sie im *RENAME*-Kommando bei einem Dateinamen eine Laufwerksangabe machen, da *RENAME* ja nur einen Namen umbenennt und nicht kopiert.

—» *RENAME (resident)*

RMAC *(CP/M 3.0, beim C128: Additional Utilities)*

Makroassembler, der verschiebbaren Code erstellt.

Syntax:

```
RMAC {Dateiname}
RMAC {Dateiname} $OPTIONEN
RMAC {Dateiname} $Rd|$Sd|$Pd
```

Beschreibung:

RMAC ist ein Makroassembler, der aus Assemblerquelldateien mit der Dateikennung *ASM* relokatiblen Code mit der Dateikennung *REL* erstellt. Dieser Code muß dann später mit *LINK* bearbeitet werden und wird zur COM-Datei.

Die im *RMAC*-Kommando angegebenen Optionen geben jeweils das Quell- oder Ziellaufwerk der entsprechenden Dateien an.

Optionen:

R	Laufwerk für REL-Datei:	A-O, Z
S	Laufwerk für SYM-Datei:	A-O, X, P, Z
P	Laufwerk für PRN-Datei:	A-O, X, P, Z

A-O	Laufwerksangabe
X	Konsolausgabe
P	Druckerausgabe
Z	Keine Ausgabe

—» *MAC, LINK, LIB*

SAVE

Speichert den Inhalt des Arbeitsspeichers in eine Datei.

Syntax:

```
SAVE
```

Beschreibung:

Das *SAVE*-Kommando kopiert den angegebenen Speicherinhalt in einer Datei. Um *SAVE* benutzen zu können, müssen Sie vor Ausführung eines Programmes, das die zu speichernde Datei einliest, *SAVE* einmal aufrufen. *SAVE* plaziert sich dann am oberen Speicherrand und ist somit dauerhaft vorhanden.

Nachdem Sie das auf *SAVE* folgende Programm verlassen, meldet sich *SAVE* automatisch sofort wieder, und Sie können den gewünschten Speicherbereich abspeichern. Dabei können Sie Start- und End-Adresse dieses Bereiches definieren.

```
A>SAVE

A>SID DATE.COM
```

Sie können nun Änderungen an der Datei vornehmen

```
#g0
```

Sie haben nun SID verlassen, und SAVE befragt Sie nach den Speichergrenzen sowie dem Filenamen.

```
SAVE Ver3.0
File (or RETURN to exit)? datum.com
Delete datum.com? Y
From? 100
To?   500
```

SET *(CP/M 3.0)*

Setzen von Datei-Attributen

Syntax:

```
SET [Optionen]
SET {d:}[Optionen]
SET {Dateiname}[Optionen]
SET {d:}{Dateiname}[Optionen]
```

Beschreibung:

Mittels des *SET*-Kommandos können Sie alle möglichen Attribute einer Datei oder einer Diskette kontrollieren und verändern. Mit die wichtigste Aufgabe von *SET* ist das *Timestamping* und die *Paßwortdefinition*. Ferner können Sie Dateien vor dem Überschreiben und Löschen schützen und vieles mehr.

Sie können mittels *SET* eine ganze Diskette ansprechen, ein Laufwerk, eine Datei oder eine Menge von Dateien. Die Attribute, die Sie mit *SET* definieren, können Sie sich mit *SHOW* anzeigen lassen.

Die Attribute:

DIR	Löscht das System-Attribut einer Datei
SYS	Setzt das System-Attribut einer Datei
RO	Datei ist *READ-ONLY*, also nur lesbar - kann nicht gelöscht werden
RW	Datei ist *READ-WRITE*; dies ist standardmäßig jede Datei. Die Datei kann dann überschrieben oder gelöscht werden.

ARCHIVE=OFF Setzt das *Archiv-Flag* auf aus. Dies bedeutet, daß die Datei nicht archiviert worden ist. PIP mit dem [A]-Attribut kann ausschließlich Dateien kopieren, die seit dem letzten Backup nicht kopiert worden sind. Durch Löschen des *Archiv-Flags* kann dieses Kopieren erzwungen werden. PIP würde nach dem Kopieren dieses Flag auf *ein* setzen.

ARCHIVE=ON Setzt das *Archiv-Flag* auf ein. Dies bedeutet, daß die Datei archiviert worden ist und beim nächsten Bachup-Kommando mit PIP und der [A]-Option nicht mehr kopiert würde, es sei denn, die Datei würde geändert oder mit *Archiv=Off* wieder zum Kopieren freigegeben.

F1=ON|OFF Setzt das vom Benutzer frei zu definierende Flag F1 auf *aus* oder *an*

F2=ON|OFF Setzt das vom Benutzer frei zu definierende Flag F2 auf *aus* oder *an*

F3=ON|OFF Setzt das vom Benutzer frei zu definierende Flag F3 auf *aus* oder *an*

F4=ON|OFF Setzt das vom Benutzer frei zu definierende Flag auf *aus* oder *an*

Paßwörter:

Ist eine Datei ohne Paßwort, so kann jeder Benutzer der Diskette, der Zugang zu SET hat (welches man ja auch schützen kann), jede beliebige Datei mit einem bis zu 8 Zeichen langen Paßwort belegen. Dieses Paßwort muß man dann immer eingeben, will man Zugriff auf diese Datei oder auf dieses mit dem Paßwort belegten Programm haben. Gibt man das Paßwort falsch ein, so wird der Zugriff auf diese Datei/Diskette

verwehrt. Will man ein Paßwort aufheben, bedienen Sie sich des zweiten Beispielkommandos *SET [PASSWORD=<CR>*. Um ein Paßwort aufzuheben, muß man das vorher definierte Paßwort eingeben. Folgende Beispiele belegen das Label, also die Diskette, mit einem Paßwort bzw. heben dies wieder auf:

```
SET [PASSWORD=SICHER]
SET [PASSWORD=<CR>
```

Um Paßwörter überhaupt auf Dateien legen zu können, muß ein entsprechendes Attribut gesetzt werden. Dies geschieht mittels:

```
SET [PROTECT=ON] bzw.
SET [PROTECT=OFF]
```

Sie müssen also ein Paßwort auf das Label setzen und das Attribut *PROTECT* auf ein setzen, erst dann können Sie Paßwörter auf einzelne Dateien legen. Mittels dem SHOW-Kommando erhalten Sie übrigens Informationen über den Paßwort-Status aller beliebigen Dateien.

Um nun eine beliebige Datei mit dem Paßwort *Geheim* beispielsweise zu belegen, geben Sie einfach ein:

```
SET demo.dem[PASSWORD=geheim]
```

Ferner kann man noch definieren, bei welchen Aktionen eine Paßwortabfrage erfolgen soll. Hierfür gibt es wiederum vier Attribute:

READ Das Paßwort wird beim Lesen, Kopieren, Schreiben, Löschen oder Umbenennen der Datei erfragt.

WRITE Das Paßwort wird beim Schreiben, Löschen oder Umbenennen der Datei erfragt. Beim Lesen der Datei wird also das Paßwort nicht benötigt. Ein sicherer Schutz also gegen unerlaubtes Löschen oder Ändern von Dateien.

DELETE Das Paßwort ist ausschließlich beim Löschen der Datei erforderlich. Kein sehr guter Schutz, da die Datei überschrieben werden könnte..

NONE Sollte ein Paßwort auf dieser Datei liegen, so wird dieses einfach ignoriert; es ist also unwirksam.

Ferner bietet CP/M 3.0 noch eine weitere Möglichkeit des Schutzes. Es kann ein sogenanntes *Default-Paßwort* definiert werden. Ist eine Datei geschützt, aber kein Paßwort definiert worden bzw. das Paßwort nicht korrekt eingegeben, so kann man *ein* Paßwort für alle geschützten Dateien definieren, auf das dann kontrolliert wird. Dies geschieht durch das Kommando:

```
SET *.COM[PASSWORT=geheim, PROTECT=READ]
SET [DEFAULT=soso]
```

Es werden also alle COM-Dateien durch das Paßwort *geheim* geschützt. Wird dieses Paßwort nicht eingegeben, so können Sie die zweite Paßwortabfrage mit *soso* quittieren, und auf diese Weise Zugriff auf die Datei erlangen.

Zeitstempeloptionen:

CREATE=ON Die Zeitstempelfunktion für Dateien wird aktiviert. Ist dieses Flag gesetzt, so wird die Erstellungszeit und das Erstellungsdatum gespeichert. Das Flag muß *vor* der Erstellung einer Datei gesetzt worden sein.

ACCESS=ON Die Zeitstempelfunktion für Dateien in Sinne des Zugriffes ist aktiviert. Es wird nun jeder Zugriff gespeichert und angezeigt. Ferner wird getrennt die letzte Veränderung der Datei (Update) gespeichert, sofern das entsprechende Flag gesetzt ist. *CREATE* und *ACCESS* schließen sich gegenseitig aus, d.h. es kann immer nur eine Funktion eingeschaltet sein. Wird *ACCESS* eingeschaltet, wird automatisch *CREATE* ausgeschaltet.

UPDATE=ON Die letzte Veränderung einer Datei *(Update)* wird mit Zeit und Datum gespeichert. Wichtig für die PIP-Operation in Verbindung mit dem [A]-Attribut.

Syntax:

```
SET [CREATE=ON]
SET [ACCESS=ON]
SET [UPDATE=ON]
```

Bemerken Sie also, daß Sie entweder *UPDATE* und *CREATE* gleichzeitig eingeschaltet haben können oder aber *UPDATE* und *ACCESS*, niemals aber *CREATE* und *ACCESS*. Sie können also kombinieren, beispielsweise:

```
SET [CREATE=ON, UPDATE=ON]
```

Diese Attribute beziehen sich *immer* auf die gesamte Diskette und es ist erforderlich, daß zuvor das Kommando *INITDIR* gegeben wurde.

—» *STAT, SHOW, SETDEF, DIR, DIRS, INITDIR*

SETDEF *(CP/M 3.0)*

Zeigen und Definieren des Suchpfades.

Syntax:

```
SETDEF
SETDEF {d:{,d:{,d:{,d:}}}}
SETDEF [DISPLAY|NODISPLAY]
SETDEF [PAGE|NOPAGE]
```

Beschreibung:

Mittels *SETDEF* können Sie sich den Suchpfad ausgeben lassen oder verändern, das Standardlaufwerk ändern und die Reihenfolge der zu suchenden Dateitypen definieren (SUB, COM).

Normalerweise wird ein transientes Programm vom Standardlaufwerk geladen bzw. von dem Laufwerk, dessen Bezeichnung Sie dem Programmnamen voranstellen. Mittels *SETDEF* können Sie nun den Suchpfad der Laufwerke angeben.

Ferner können Sie mit dem *SETDEF*-Kommando sich den Suchpfad auf dem Bildschirm ausgeben lassen. Es wird Ihnen dann die Laufwerksreihenfolge ausgegeben, sowie das aktuelle Standardlaufwerk und die Reihenfolge der Dateitypen, nach denen gesucht wird: (COM, SUB) oder (SUB, COM).

```
SETDEF [TEMPORARY=d:]
```

definiert das Standardlaufwerk für Temporärdateien. Das Standardlaufwerk für Temporärdateien ist das System-Standardlaufwerk.

```
SETDEF {d:{,d:{,d:{,d:}}}}
```

Mit diesem Kommando definieren Sie die Reihenfolge der anzusprechenden Laufwerke.

```
SETDEF B:,*
```

beispielsweise definiert, daß zunächst auf Laufwerk B: gesucht wird, erst dann auf dem Standardlaufwerk.

```
SETDEF [ORDER=(typ{,typ})]
```

Sie können angeben, ob bei Eingabe eines Kommandos zunächst nach Dateien mit der Endung SUB (also Submit-Dateien) gesucht werden soll oder nach COM-Dateien.

Mittels *SETDEF [NODISPLAY | DISPLAY]* können Sie wählen, ob Informationen ausgegeben werden sollen, wenn ein Kommando ausgeführt wird, beispielsweise:

```
B>PIP
A:PIP      COM
CP/M 3 PIP Version 3.0
*
```

bei eingeschaltetem *DISPLAY*, so sieht es bei ausgeschaltetem *DISPLAY* aus:

```
B>PIP
*
```

Mittels Einschalten von *PAGE* können Sie definieren, daß CP/M 3.0 nach jedem Füllen eines Bildschirmes anhält, so daß erst nach Quittieren mit einer Taste der Bildschirm scrollt. So kann man vermeiden, daß Informationen durch Scrollen verloren gehen.

SETUP *(CP/M 2.2, Schneider)*

Startkonfiguration des Systems ändern.

Syntax:

```
SETUP
```

Beschreibung:

Mittels *SETUP* können Sie sich eine private Version Ihres CP/M 2.2 erstellen.. Sie können die Anfangsfarben des CP/M ebensogut ändern wie die Tastaturbelegung oder das erste automatisch zu sendende Kommando. Dieses Kommando existiert ausschließlich auf der CP/M-Version 2.2 für die Schneider-Rechner.

Bei der Definition der Farben etc. bedienen Sie sich der im Schneider CPC üblichen Kontrollcodes.

Die erste Frage des SETUP ist, ob Sie den sogenannten *Initial Command Buffer* ändern wollen. Wollen Sie dies tun, so antworten Sie hier mit *Y* für Ja.

```
Enter new initial command buffer:_
```

Wollen Sie nun beispielsweise, daß nach dem Booten des Systems zunächst das aktuelle Inhaltsverzeichnis der Diskette ausgegeben wird, so müssen Sie in den Initial Command Buffer das Kommando *DIR* inklusive einem *<CR>* schreiben. Exemplarisch für die anderen Abfragen erreichen Sie dies, indem Sie

```
DIR^M
```

eingeben. Geben Sie keineswegs CTRL-M ein, dies würde ja nur <CR> simulieren. Betätigen Sie die Pfeil-Hoch-Taste und die M-Taste getrennt. Bei der Ausführung des Initial Command Buffers wird dies schon berücksichtigt.

Nachdem Sie dies eingegeben haben, können Sie auch selbst <CR> betätigen und auf die Frage, ob der Initial Command Buffer nun korrekt sei, auch mit *Y* für Ja antworten.

Unter *Sign-on-String* dürfen Sie die Zeichen verstehen, die unmittelbar nach dem Booten gedruckt werden; dies sind Steuerzeichen für 80-Zeichen-Modus, Farbe etc.

Hiernach wird es Ihnen ermöglicht, die Tastatur umzubelegen bzw. neue Tasten zu definieren. Dieser Menüpunkt nennt sich *Keyboard Translation*. Hier erfolgt die Eingabe parallel zum BASIC-Kommando *KEY DEF*. Die Funktion *Keyboard Expansions* wird genauso programmiert wie das BASIC-Kommando *KEY*.

Als vorletzte Frage können Sie sich entscheiden, ob die gemachten Änderungen abgespeichert werden sollen. Haben Sie dies mit Ja bestätigt, können Sie - wenn gewünscht - CP/M noch einmal neu starten, um die Konsequenzen Ihrer Änderungen direkt am Bildschirm verfolgen zu können.

SHOW *(CP/M 3.0)*

Zeigt die Daten einer Diskette an.

Syntax:

```
SHOW [Option]
SHOW {d:}[Option]
SHOW {d:}{[SPACE|LABEL|USERS|DIR|DRIVE]}
```

Beschreibung:

Mittels *SHOW* können Sie sich wichtigsten vorhandenen Daten einer Diskette auf dem Bildschirm ausgeben lassen. Dabei erhalten Sie Informationen über die folgenden Punkte:

- *Zugriffsmodus*
- *Freier Speicherplatz*
- *Aktueller Userbereich*
- *Diskettenlabel (-name)*
- *Anzahl der verbleibenden Disketteneinträge*
- *Anzahl der Einträge pro Userbereich*
- *Laufwerksdaten*

SHOW ist als sinnvolle Ergänzung zu *DIR* zu sehen und auch so zu verwenden. Mittels *SHOW* können Sie beispielsweise auch erfahren, ob einzelne Dateien oder gar die gesamte Diskette mit einem Schutzwort versehen worden ist.

Mittels *SHOW* ohne Parameter erhalten Sie Auskunft über alle angegebenen Punkte. Geben Sie jedoch Parameter an, so werden Ihnen lediglich Informationen über diese gewünschten Punkte ausgegeben.

—» *SET, SETDEF, STAT*

SID *(CP/M 3.0, C128 Additional Utilities)*

Debuggerprogramm des CP/M.

Syntax:

```
SID
SID {Dateiname}
SID {Dateiname}{,SYM-Dateiname}
```

Beschreibung:

Mit SID *(Symbolic Instruction Debugger)* können Sie COM- oder HEX-Dateien in den Arbeitsspeicher laden, dort ansehen und verändern. Sie können auch eine entsprechende Symboltabelle mitladen und haben somit mehr Informationen parat. *SID* kann auch ablauffähige Programme in INTEL-8080-Mnemonics umwandeln. Möglich ist: Setzen von Real-Time-Breakpoints, voll-monitor-unterstützte Ausführung, symbolische Disassemblierung, Assemblierung, Speicherauslistung und vieles mehr.

Leider befindet sich auch dieses Programm beim C128 ausschließlich auf der Additional-Utility-Diskette und steht somit nur Käufern dieses Paketes zur Verfügung. *SID* ist übrigens eine Weiterentwicklung des *DDT*-Programmes unter CP/M 2.2.

SID ist wohl für jeden Assemblerprogrammierer ein *Muß*, da er durch die Möglichkeit der Breakpoints eine sinnvolle und hilfreiche Möglichkeit der Programmablaufkontrolle bietet.

Kommandos:

As	Assemblieren der nachfolgenden Kommandos. *s* ist die Startadresse.
Cs{b{,d}}	Aufruf eines Maschinenspracheunterprogrammes von *SID* aus. Hierbei ist *s* die Startadresse, *b* die optionale Angabe

des BC-Registerpaares und *d* die optionale Angabe des DE-Registerpaares.

D{W}{s}{,f} Ausgabe von Speicherbereichen in Hexadezimal und ASCII. *W* bedeutet Ausgabe in 16-Bit-Format. *s* ist die Startadresse und *f* die Endadresse.

En{,s} Laden der Datei mit Namen *n*. Optional wird auch die Symboltabelle mit Namen *s* geladen.

E*s Laden einer Symboltabelle ausschließlich (ohne Datei).

Fs,f,d Füllen eines Speicherbereiches. *s* ist die Startadresse, *f* die Endadresse und *d* das Zeichen, mit dem der Speicherbereich aufgefüllt werden soll.

G{p}{,a{,b}} Programm starten. *p* ist hierbei die Startadresse; *a* ist ein temporärer Unterbrechungspunkt. An dieser Stelle wird also angehalten. *b* ist hier ein zweiter temporärer Unterbrechungspunkt. *G0* veranlaßt SID, einen Warmstart auszuführen.

H Es wird die Symboltabelle in Hexadezimal, in Dezimal und in ASCII ausgegeben.

I Eingabe eines CCP-Kommandos

L{s}{,e} Auslisten des Speicherbereiches ab *s* bis *e* in 8080-Mnemocode.

Ms,e,z Bewegen eines Speicherbereiches. Hierbei wird der Bereich von *s* bis *e* nach *z* kopiert.

Rd{,o}	Liest die Datei in den Speicher (oder Symboltabelle). *o* gibt hierbei eventuell den Offset an, an den geladen werden soll.
S{W}w	Schreiben von Werten an eine Adresse *w*. Ist *W* angegeben, so werden 16-Bit-Worte geschrieben.
T{W}{n{,a}}	Trace-Modus von SID. *W* gibt an, daß Unterroutinen nicht getracet werden sollen (Sehr nützlich). *n* gibt die eventuelle Anzahl der zu tracenden Schritte an und *a* die Startadresse.
U{W}{n{,a}}	Bedingte Programmausführung ohne TRACE. Dabei kann die Startadresse *a* angegeben werden, sowie die Anzahl der auszuführenden Schritte *n* und ob Unterroutinen getracet werden sollen *W*.
V	Anzeige des nächsten verfügbaren Speicherplatzes (NEXT), des Program Counter (PC) sowie die Endadresse des verfügbaren Speicherplatzes (END).
Wd{,s{,e}}	Schreibt den Speicherbereich von *s* bis *e* in die Datei mit Namen *d*.
X{f}{r}	Untersucht das Flag *f* (C, E, I, M oder Z) des Registers *r* (A, B, D, H, P oder S).

—» DDT, MAC, RMAC, SAVE, LINK, LIB

SUBMIT

Ausführen einer Reihe von Systemkommandos.

Syntax:

```
SUBMIT
SUBMIT {Dateiname.SUB}
SUBMIT {Dateiname} {Para1} {Para2} {Para3}...{Para10}
```

Beschreibung:

Normalerweise geben Sie jedes Kommando einzeln über Tastatur ein. Haben Sie häufig wiederkehrende Eingabesequenzen, beispielsweise bei der Programmierung in Assembler, so können Sie mittels der *SUBMIT*-Dateien diese Sequenzen unter einem Namen zusammenfassen; mit *SUBMIT* wird diese Sequenz dann an einem Stück ausgeführt.

Nach dem Starten des Systems (Booting) wird nach der Datei *PROFILE.SUB* gesucht. Kann diese Datei gefunden werden, so wird sie auch automatisch ausgeführt (gilt nur für CP/M 3.0). Auf diese Weise können Sie eine Befehlsfolge, die nach dem Starten des Systems in irgend einer Form obligatorisch ist, automatisch ablaufen lassen.

Am einfachsten ist es, Sie schreiben Ihre *SUBMIT*-Datei mit einer Textverarbeitung (beispielsweise ED). *SUBMIT*-Dateien müssen die Dateikennung *SUB* besitzen.

Ferner kann man Parameter an diese *SUBMIT*-Dateien übergeben, so daß man sich also auch in der *SUBMIT*-Datei nicht festlegen muß. Die Parameter werden von 1 bis 0 durchnumeriert und durch ein führendes Dollarzeichen in die *SUBMIT*-Dateien eingebaut:

```
DIR *.txt
ERA $1.txt
DIR $1
SHOW $2
```

würde beispielsweise bei einem Aufruf durch

```
SUBMIT datei alpha *.txt
```

zunächst die TXT-Dateien auslisten. Dann würde die eingegebene TXT-Datei gelöscht und nochmals zur Kontrolle ausgegeben, sie könnte ja löschgeschützt gewesen sein. Danach wendet sich diese *SUBMIT*-Datei dem zweiten Parameter zu, der an das SHOW-Programm übergeben wird.

Wollen Sie übrigens ein Dollarzeichen im Text haben, so ist es erforderlich, zwei Dollarzeichen hintereinander zu stellen.

Es kann vorkommen, beispielsweise bei PIP oder SID, daß Programmeingaben gefordert sind. Auch diese können Sie über SUBMIT-Dateien übergeben. Dies sieht dann so aus:

```
SID
<Edatei
<L0100,0200
<G0
DIR *.TXT
```

Hier würde beispielsweise zunächst SID gestartet. Dann würde die Datei *datei* geladen, dann der Bereich 0100 bis 0200 ausgelistet und dann die Kontrolle wieder zurückgegeben; danach wird dann noch ein Disketteninhaltsverzeichnis der TXT-Dateien ausgegeben.

—» *GET, PUT*

STAT *(CP/M 2.2, CP/M 86)*

Ausgabe von Dateiattributen.

Syntax:

```
STAT
STAT {Dateiname}
STAT DEV:
STAT DSK:
STAT USR:
STAT {Dateiname} $SYS
```

Beschreibung:

Mittels *STAT* können Sie sich zusätzliche Informationen über Disketten und Dateien ausgeben lassen, die Ihnen das *DIR*-Kommando nicht ausgibt. Dies sind beispielsweise Angaben wie Dateigröße in Records (ein Record ist eine CP/M-Größe und bedeutet 128 Byte), Ausgabe von Systemdateien, Ausgabe aller benutzten Benutzerbereiche und vieles mehr.

STAT gibt den Zustand aller im System derzeit aktiven Laufwerke in einer Liste wie folgt aus:

```
{Laufwerk}: {Schutz}, Space: {n}k
```

Wollen Sie wissen, welche Benutzerbereiche Dateien enthalten, so liefert Ihnen das Kommando

```
STAT USR:
```

entsprechende Hinweise. Dies sieht dann auf dem Bildschirm etwa so aus:

```
Active User : 5
Active Files: 0 1 3 5 7
```

Eine Ausgabe der Zuordnung von logischen und physikalischen Geräten erhalten Sie durch Eingabe des folgenden Kommandos:

```
STAT DEV:

CON: is CRT:
RDR: is TTY:
PUN: is TTY:
LST: is LPT:
```

Durch *STAT {Kanal}={Gerät}* können Sie jedem Kanal ein entsprechendes vorhandes Gerät zuordnen. Beispiele hierfür sind:

```
STAT CON:=CRT:
STAT LST:=TTY:
```

Um die Bedeutung der verschiedenen Bezeichnungen für logische und physikalische Geräte zu vergleichen, sehen Sie unter dem CP/M-Plus-Kommando *DEVICE* nach, da dieses ähnlich wie *STAT* mit denselben Bezeichnungen arbeitet

Optionen:

$R/W Setzen des Schreib-/Lesemodus. Es wird der Nur-Lese-Status für das Laufwerk/die Datei aufgehoben. Die Datei kann jetzt auch wieder gelöscht werden.

$R/O Setzen des Nur-Lese-Status. Die Datei kann nun nicht mehr gelöscht oder überschrieben werden. Allerdings kann die Datei weiter gelesen werden.

$DIR Setzen des Directory-Modus. Die Datei gehört nicht der Gruppe der Systemdateien an und wird im Inhaltsverzeichnis ausgegeben.

$SYS Setzen des Systemstatus. Die Datei wird beim Auslisten des Inhaltsverzeichnisses mit DIR nicht mehr ausgegeben. Die auf diese Weise markierten Dateien können nur noch mit den STAT-Kommando ausgegeben werden.

Die Optionen werden nicht in eckigen Klammern hinter dem Dateinamen angegeben, sondern mit führendem Dollarzeichen. Um eine Datei also beispielsweise nur-lesbar zu machen, ist folgendes Kommando notwendig:

```
STAT dump.com $R/O
```

Diskettendaten:

Disketteninformationen sind beispielsweise, wieviel Platz sich noch auf der Diskette befindet etc. Diese Informationen erhalten Sie:

```
STAT a:DSK:

   A: Drive Chacteristics
1368: 128 Byte Record Capacity
 171: Kilobyte Drive Capacity
  64: 32 Byte Directory Entries
  64: Checked Directory Entries
 128: Records/ Extent
   8: Records/ Block
  36: Sectors/ Track
   2: Reserved Tracks
```

Werte (Values):

Als letztes bietet sich noch die Möglichkeit, alle STAT-Werte ausgeben zu lassen - Sie können in dieser Liste eine Art *Erinnerung* sehen, was mit STAT alles möglich ist:

```
STAT VAL:

Temp R/O Disk: D:=R/O
Set indicator: d:filename.typ $R/O $R/W $SYS $DIR
Disk Status  : DSK: d:DSK:
User Status  : USR:
IObyte Assign:
CON: = TTY: CRT: BAT: UC1:
RDR: = TTY: PTR: UR1: UR2:
PUN: = TTY: PTP: UP1: UP2:
LST: = TTY: CRT: LPT: UL1:
```

—» *DEVICE, SHOW, DIR, DIRSYS*

SYSGEN *(CP/M 2.2)*

Kopiert das System CP/M 2.2 auf Diskette.

Syntax:

```
SYSGEN
SYSGEN {Dateiname}
SYSGEN *
```

Beschreibung:

SYSGEN schreibt das Ergebnis eines *MOVCPM*-Kommandos auf die Systemspuren einer beliebigen Zieldiskette. *SYSGEN* ist also nur in Verbindung mit *MOVCPM* sinnvoll. *SYSGEN** schreibt dabei das unmittelbar zuvor durch *MOVCPM* erstellte System auf Diskette (Systemspuren). *SYSGEN* in Verbindung mit einem Dateinamen sichert das durch den Dateinamen angegebene System auf die Systemspuren einer Zieldiskette. Wird kein Dateiname und kein Stern angegeben, so wird nach Quell- und Zieldiskette gefragt.

Mit *MOVCPM* kann man CP/M im Speicher beliebig verschieben, manchmal wird dies notwendig. Sie können CP/M in Schritten von 256 Byte verschieben. Die Position im Speicher bzw. die Größe der *TPA (Transient Program Area)* - das ist der für Programme und Daten verfügbare Speiherplatz - wird in Werten von 64 bis 179 angegeben.

—» *MOVCPM*

TYPE *(CP/M 3.0)*

Zeigt den Inhalt einer Datei auf dem Bildschirm an.

Syntax:

```
TYPE
TYPE {d:}{Dateiname}
TYPE {d:}{Dateiname}{[NOPAGE|PAGE]}
```

Beschreibung:

Neben dem transienten *TYPE* gibt es noch das residente *TYPE*. Das transiente *TYPE* wird nur dann geladen, wenn die Option angegeben oder kein Dateiname angegeben wird.

Wird kein Dateiname angegeben, so meldet sich *TYPE* mit:

```
Enter filename:
```

Sie können die Ausgabe auf den Bildschirm mit CTRL-S und CTRL-Q beliebig anhalten und fortführen. Haben Sie die Option *NOPAGE* angegeben, so wird kontinuierlich auf dem Bildschirm ausgegeben und ohne anzuhalten gescrollt. Anders bei Angabe der Option *PAGE*; hier wird, sobald der Bildschirm gefüllt wurde, die Ausgabe angehalten. Erst wenn Sie eine beliebige Taste betätigt haben, wird mit der Ausgabe fortgefahren.

Haben Sie einen Dateinamen angegeben, der nicht existiert, meldet TYPE:

```
No File
```

Um die Ausgabe der Datei auf den Drucker umzuleiten, müssen Sie bevor Sie die Datei mittels *TYPE Dateiname* ausgaben lassen, den Protokollmodus auf den Drucker mit CTRL-P einschalten.

XREF *(CP/M 3.0, C128 Additional Utilities)*

Erstellt eine Referenzliste der verwendeten Symbole.

Syntax:

```
XREF {d:}{Dateiname}
XREF {d:}{Dateiname} {$P}
```

Beschreibung:

Die Assembler *MAC* und *RMAC* erstellen bei der Assemblierung von Quelldateien eine Symboltabelle, die mit der Dateikennung *SYM* abgespeichert werden kann. *XREF* lädt nun die erstellte Datei und bearbeitet die Symboltabelle. *XREF* erstellt unter dem Dateinamen <Dateiname>.XRF eine Assembler-Liste (ähnlich dem <Dateiname>.PRN) und hängt hieran noch eine ausführliche Auflistung aller vorkommenden Symbole inklusive Angabe der Referenzen im Programm unter Angabe der Zeilennummer an. Man nennt so etwas auch *Cross Reference List*. Die Amerikaner schreiben für Cross gerne nur kurz den Buchstaben X, weshalb dieses Kommando *XREF* heißt.

Durch Angabe der Option $P können Sie erzwingen, daß die Ausgabe direkt zum Drucker gesendet wird.

—» *MAC, RMAC, LINK*

Fehlermeldungen und Tastaturkontrollzeichen

Fehlermeldungen

Fehlermeldungen können überall auftauchen: im *CCP (Console Command Processor)*, also nachdem ein Kommando eingegeben wurde, aber vor allem innerhalb von Programmen.

Um Ihnen die möglichen Fehler verständlicher zu machen, hier eine Auflistung der Fehlermeldungen und einen kurzen Kommentar, was Ihnen die Fehlermeldung sagen will.

Wundern Sie sich dabei bitte nicht, wenn manche Fehlermeldung in verschiedenen Schreibweisen abgedruckt sind. Dies liegt daran, daß alte CP/M-Versionen auf Rechner liefen, die nicht in der Lage waren, Umlaute oder Großbuchstaben auszugeben. Bei den neueren CP/M-Versionen konnte man aber auch in Großbuchstaben ausdrucken, so daß man diese dann auch in den Fehlermeldungen verwenden konnte. Auch gibt es manchmal mehrere Fehlermeldung mit derselben Bedeutung; dies liegt ebenfalls an den verschiedenen Versionen.

Assign a password to this file.

Es wurde bei SET versucht, den Paßwort-Modus ohne entsprechendes Paßwort einzuschalten.

Auxiliary device redirection not implemented

AUXIN oder AUXOUT konnte nicht in eine Datei umgeleitet werden.

Bad close.

Bei *SAVE* ist während des Versuches, die Datei zu schließen, ein Fehler aufgetreten. Wahrscheinlich ist die Datei schreibgeschützt.

Bad Logical Assignment;

Nur folgende logische Einheiten sind erlaubt: CONIN:, CONOUT:, AUXIN:, AUXOUT:, LST:.

BAD PARAMETER

Falsch eingegebener Parameter im PIP-Programm. Geben Sie den Parameter neu ein.

Bad password.

Während *RENAME* wurde ein falsches Paßwort eingegeben. Die Ausführung wird nicht zugelassen.

Bank one not allowed.

Bank 1 kann bei *GENCPM* während der Systemerstellung nicht als verfügbar definiert werden.

Baud rate cannot be set for this device.

Nur bei Geräten mit softwaremäßig feststellbaren Baudraten kann eine andere Baudrate programmiert werden. Um dies festzustellen, bedienen Sie sich des DEVICE-Kommandos.

Break "x" at c

Dieser Fehler tritt nur bei *ED* auf. *x* steht für eines der folgenden Zeichen, *c* ist die Stelle, an der der Fehler aufgetreten ist.

#	Fehler beim Suchen. ED kann String nicht finden.
?	Kommando c ist unbekannt
O	Die Datei, die im R-Kommando angegeben wurde, kann nicht gefunden werden
>	Der Buffer ist voll. *ED* kann nicht mehr Zeichen in den Buffer schreiben, oder den definierte String in einem der Kommandos F, N oder S ist zu lang
E	Kommando abgebrochen. Ein Tastendruck an der Konsole hatden Befehl abgebrochen
F	Diskette oder Directory ist voll.

CANNOT CLOSE:
Cannot close file.
CANNOT CLOSE DESTINATION FILE - <Filename>

Fehler kann auftreten bei *PIP, RMAC, MAC, LIB, HEXCOM, GENCOM* oder *SUBMIT*. Eine Ausgabedatei konnte nicht geschlossen werden, beispielsweise weil die Diskette während der Arbeit gewechselt worden ist.

Cannot delete file.

CP/M kann eine Datei nicht löschen. Vergewissern Sie sich, daß die Datei nicht Read-Only oder paßwortgeschützt ist.

Cannot have both create and access time stamps.

Beim SET-Kommando dürfen Sie nicht das ACCESS- und das CREATE-Attribut gleichzeitig setzen.

Cannot label a drive with a file referenced.

SET erlaubt es nicht, Laufwerks- und Dateibezeichnungen zu mischen.

CANNOT OPEN SOURCE FILE

Bei *HEXCOM* war die HEX-Datei nicht auf dem abgegebenen Laufwerk.

Cannot redirect from BIOS.

Tritt bei PUT oder GET nur dann auf, wenn das System ein falsches BIOS hat.

Cannot set both RO and RW.

Eine Datei kann beim SET-Kommando nicht Read-Only und Read-and-Write definiert werden.

Cannot set both SYS and DIR.

Eine Datei kann nicht das SYS- und das DIR-Attribut gleichzeitig haben.

CAN'T DELETE TEMP FILE

Eine temporäre Datei (Endung: $$$) kann nicht gelöscht werden, weil beispielsweise Read-Only vorliegt.

CHECKSUM ERROR.
checksum error

Ein Prüfsummenfehler ist bei einem HEX-Record (im Header) aufgetreten. Der Record oder die gesamte Datei muß korrigiert werden.

Close error.

Die Diskette wurde bei XREF vor Abschluß des Programmes entfernt.

Close operation failed.

Ein Problem ist beim Schließen einer Datei am Ende von COPYSYS aufgetreten.

Closing file HELP.DAT
Closing file HELP.HLP

HELP hat während Schließen einer der beiden Dateien Schwierigkeiten.

COM file found and NULL option.

Die NULL-Option bei GENCOM gibt an, daß keine COM-Dateien, sondern lediglich RSX-Dateien geladen werden.

.COM file required

Durch Angabe von Optionen muß das residente DIR, RENAME, TYPE oder ERASE auf das entsprechende transiente auf Diskette zugreifen, findet diese Datei aber nicht.

CORRECT ERROR, TYPE RETURN OR CTRL-Z

Mit PIP sollte eine HEX-Datei transferiert werden, die einen Checksummen-Fehler aufweist. Der entsprechende Record oder die gesamte Datei muß neu eingerichtet werden.

CPMLDR error: failed to open CPM3.SYS

Die CP/M-Plus-Datei CPM3.SYS kann nicht gefunden werden.

CP/M Error on d: Disk I/O
BDOS Function = xx File = <Dateiname>

CP/M meldet eine der oben genannten Meldungen, wenn das angesprochene Laufwerk nicht existiert, ausgeschaltet ist, die Tür offen ist oder sich keine Diskette im Laufwerk befindet.

CP/M Error on d: Read/Only Disk
BDOS Function = xx File = <Dateiname>

Sie wollten eine Datei löschen, umbenennen oder verändern, haben aber die gesamte Diskette auf den Read-Only-Status gesetzt. Sie müssen mit *SET* den Status des Labels ändern.

CP/M Error on d: Read/Only File
BDOS Function = xx File = <Dateiname>

Es wurde versucht, eine bestimmte Datei zu löschen, umzubenennen oder zu verändern, obwohl das Read-Only-Attribut auf dieser Datei liegt. Mit *SET* muß diese Datei auf Read-Write umgestellt werden.

Date and Time Stamping Inactive.

Die Datumsfunktion für Dateien wurde eingeschaltet, jedoch wurde das Inhaltsverzeichnis der Diskette nicht mit *INITDIR* auf diese Tatsache vorbereitet.

DESTINATION IS R/O, DELETE (Y/N)?

Die bei *PIP* angegebene Zieldatei existiert bereits und ist Read-Only. Bestätigen Sie mit Y, so wird die alte Datei überschrieben; wollen Sie dies nicht, so bestätigen Sie mit N, Sie erhalten dann die Meldung **NOT DELETED**.

Device Reassignment Not Supported.
Enter new assignment or hit RETURN.

Eine Zuweisung bei *DEVICE* ist nicht erlaubt (Gerät existiert nicht).

Directory already re-formatted.

Sie haben versucht, mittels *INITDIR* eine Diskette auf das Timestamping vorzubereiten; dies wurde jedoch bereits gemacht.

Directory full

Bei der letzten Funktion wurde festgestellt, daß nicht genügend Platz auf der Diskette ist. Um beispielsweise bei *ED* nicht alle Daten zu verlieren, empfiehlt es sich, eine andere Diskette einzulegen, auf der genügend Platz ist.

Directory needs to be reformatted for date/time stamps.

Es wurde mittels *SET* definiert, die Dateien mit Zeitstempeln zu versehen, jedoch wurde das Inhaltsverzeichnis der Diskette noch nicht mit *INITDIR* behandelt.

DISK FULL

Diese Meldung gibt Ihnen *ED*, wenn vor dem Speichern einer Datei nicht genügend Platz auf der Diskette ist.

DISK READ
DISK READ ERROR:
Disk read error: <Dateiname>
DISK READ ERROR - <Dateiname>

Die gewünschte Datei kann nicht gelesen werden; sie existiert zwar, die Diskette ist aber beispielsweise zerstört.

DISK WRITE.
Disk Write Error
DISK WRITE ERROR: <Dateiname>
DISK WRITE ERROR - <Dateiname>

Eine Schreiboperation konnte nicht durchgeführt werden, weil beispielsweise die Diskette voll oder an einer Stelle schadhaft ist.

Do you want another file? (Y/N)

Diese Abfrage erhalten Sie bei *PUT*. Wollen Sie die Bildschirmausgaben zusätzlich auf eine Datei, quittieren Sie mit Y, sonst mit N.

Drive defined twice in search path

Sie haben bei *SETDEF* bei der Pathangabe zweimal dasselbe Laufwerk angegeben.

Drive Read Only

Sie haben versucht eine Datei zu löschen, umzubenennen oder zu verändern, die sich auf einem mit dem Attribut Read-Only versehenen Laufwerk befindet.

Drive specified has not been defined.

Es sind keine Buffer für dieses Laufwerk vorgesehen, da es bislang noch nicht definiert worden ist.

Duplicate RSX in header. Replacing old by new.
This file was not used.

Das angegebene RSX ist bereits an eine COM-Datei gebunden.

Duplicate inputs RSX.

Zwei oder mehr RSX vom gleichen Namen sind definiert worden. GENCOM kann nur eine davon nutzen.

Equals (=) delimiter missing at line <nn>.

Das Gleichheitszeichen (=) fehlt in der angegebenen Zeile.

END OF FILE, ^Z, ?

PIP ist unerwartet auf eine EOF-Marke während eines Hextransfers gestoßen.

End of line expected.

Es sind zu viele Parameter angegeben worden, und ein Ende der Zeile wurde erwartet. Alles überflüssige wird ignoriert.

Error at end of line:

Der gefundene Fehler befindet sich am Ende der Zeile.

Error on line <nnnnn>:

Dieser Fehlermeldung folgt eine weitere fehlerbestimmende Meldung. Diese Meldung teilt Ihnen lediglich mit, in welcher Zeile einer SUBMIT-Datei sich der Fehler befindet.

FILE ERROR

Die Diskette oder das Inhaltsverzeichnis ist voll, ED kann nichts mehr auf die Diskette schreiben.

File already exists; Delete it(Y/N)

Bei *PUT* oder *RENAME* soll eine Datei überschrieben werden, die bereits existiert. Sie können dies entweder durch Y bestätigen oder durch N verhindern.

File cannot fit into GENCPM buffer: <Dateiname.SPR>

Es ist nicht genügend Speicher vorhanden, um ein System zu generieren.

File exists, erase it

Die bei *ED* angegebene Datei existiert bereits. Entweder wechseln Sie die Diskette oder löschen diese Datei.

FILE IS READ/ONLY
File is Read Only

Die Datei, die bei *ED* oder *PUT* überschrieben werden soll, ist durch Read-Only geschützt.

FILE NAME ERROR:

Der bei *LIB* eingegebene Quelldateiname ist nicht gültig

File not found.
FILE NOT FOUND - <Dateiname>

Eine definierte Quelldatei existiert nicht (*DUMP*, *ED*, *GENCOM*, *GET*, *PIP* oder *SET*). Überprüfen Sie, ob Sie die korrekte Diskette eingelegt bzw. das richtige Laufwerk angesprochen haben.

First submitted file must be a COM file.

Die erste im GENCOM-Kommando angegebene Datei muß vom Typ COM sein, es sei denn, die NULL-Option ist angegeben.

HELP.DAT not on current drive.

HELP kann die angegebene Datei nicht auf dem Standardlaufwerk finden.

Illegal command tail

Bei *DIR* wurden falsche Optionen angegeben oder die Syntax der Optionen wurde nicht korrekt eingehalten.

Illegal Format Value.

Ausschließlich die Optionen SIZE und FULL können als Anzeige-Formate angegeben werden.

Illegal Global/Local Drive Mixing.

Sie haben die DRIVE-Option benutzt und denoch im Dateinamen eine Laufwerksangabe gemacht.

Illegal filename.

Bei SAVE wurde ein syntaktisch nicht einwandfreier Dateiname angegeben.

Illegal Option or Modifier.

Eine ungültige Option oder eine unzulässige Abkürzung wurde verwendet.

Illegal date/time specification

Das Format des angegebenen Datums oder der angegebenen Zeit ist nicht korrekt.

Incorrect file specification

Der bei *RENAME* angegebene Dateiname entspricht nicht den Syntaxregeln.

Insufficient Memory
INSUFFICIENT MEMORY:

Es ist nicht genügend Platz vorhanden, um *GET*, *LINK*, *PUT* oder *SUBMIT* aufzurufen; *SUBMIT* wurde zu häufig verschachtelt.

Invalid ASCII character

Die SUBMIT-Datei beinhaltet einen unzulässigen Code (0FFH).

Invalid character at line <nn>.

Das eingegebene Zeichen muß eine Ziffer sein.

Invalid command

Der in *PUT* oder *GET* eingegebene String konnte nicht als korrektes Kommando erkannt werden.

Invalid delimiter.

Bei *DEVICE*, *GET*, *PUT* oder *SETDEF* waren die Anführungszeichen, die Klammern [ ], = oder <Leer> an dieser Stelle nicht zulässig.

INVALID DESTINATION:

Bei *PIP* wurde ein unbekanntes Laufwerk oder ein unbekanntes Gerät angesprochen.

INVALID DIGIT - <Dateiname>

Ein unerlaubter HEX-Wert wurde bei PIP in Verbindung mit einer HEX-Datei gefunden.

Invalid drive ignored at line <nn>.
Invalid drive.

Laufwerksbezeichnung unerlaubt. Bei *SETDEF* sind die Bezeichnungen * sowie A bis P erlaubt, sonst lediglich A bis P.

Invalid drive name (Use A, B, C oder D)

COPYSYS, *SYSGEN* etc. gestatten nur die Laufwerksbezeichnungen A, B, C und D.

INVALID FORMAT

Das eingegebene PIP-Kommando ist in der Form nicht erlaubt. Sehen Sie hierzu die Syntax von *PIP* nach.

INVALID HEX DIGIT.

Während der Erstellung einer COM-Datei hat *HEXCOM* einen unerlaubten HEX-Wert vorgefunden. Die HEX-Datei muß korrigiert werden

Invalid number

Bei *DEVICE* wurde eine Nummer erwartet, aber nicht gefunden oder die Nummer war außerhalb des gültigen Bereiches von 0 bis 255

Invalid option.

DEVICE, *GET*, *SETDEF:* Eine eingegebene Option war nicht korrekt oder nicht erlaubt

Invalid option or modifier.

Die bei *DIR*, *GET* oder *PUT* angegebene Option ist nicht bekannt.

INVALID PARAMETER:

Die bei *MAC* oder *RMAC* angegebene Option ist nicht erlaubt. Die Eingabezeile wird auf dem Bildschirm bis zur Fehlerstelle wieder ausgegeben.

INVALID PASSWORD
Invalid password or passwords not allowed.

Bei *ED* oder *PIP* wurde ein falsches Paßwort eingegeben oder es wurde ein Paßwort angegeben, aber die Datei war gar nicht paßwortgeschützt.

Invalid physical device.

Das angegebene Gerät im DEVICE-Kommando ist nicht bekannt.

Invalid RSX type

Der Dateityp muß RSX sein.

Invalid SCB offset.

Der bei *GENCOM* angegebene SCB ist außerhalb der Grenzen von 00H bis 64H.

INVALID SEPERATOR

Es wurde ein unerlaubtes Trennzeichen zwischen zwei Dateinamen eingefügt.

INVALID SOURCE

Es wurde bei *PIP* ein unerlaubtes Gerät oder Laufwerk angesprochen. Erlaubte Geräte sind AUX(IN/OUT) und CON(IN/OUT).

Invalid type for ORDER option.

Die angegebene Dateikennung für den Suchpfad ist nicht COM oder SUB.

Invalid SYM file format

Der für die Symboltabelle eingegebene Dateiname ist nicht gültig (XREF).

INVALID USER NUMBER

Sie haben beim PIP-Kommando und der G-Option einen unerlaubten Benutzerbereich angegeben. Erlaubte Benutzerbereiche nur von 0 bis 15.

Invalid wildcard.

Die bei *RENAME* verwendeten Joker (wild cards) sind in dieser Form nicht gültig oder sind falsch plaziert.

Invalid wild card in the FCB name or type field.

GENCOM erlaubt keinerlei Joker (wild cards) im Dateinamen.

LOAD ADDRESS LESS THAN 100.

Die Programmstartadresse des von HEXCOM zu bearbeitenden Programmes liegt unterhalb der Grenze von 0100H. Es kann folglich keine COM-Datei erstellt werden. Überprüfen Sie die Startadresse Ihres Programmes.

MAIN MODULE ERROR:

Es wurde beim Linken unerlaubterweise ein zweites Hauptmodul gefunden.

Make error

Es ist für XREF nicht mehr genügend Platz im Inhaltsverzeichnis.

Memory conflict - xxx

Speicherbereiche überlappen sich.

MEMORY OVERFLOW:

Speicherplatz reicht nicht aus, um die LINK-Ausführungen zu Ende zu führen.

Minimum number of buffer is 1.

Das erste Laufwerk muß zumindest einen Buffer zugeordnet haben.

Missing Delimiter or Unrecognized Option.

Beim ERASE-Kommando wurden unerlaubte Zeichen hinzugefügt.

Missing parameter variable at line <nn>.

In der angegebenen Zeile fehlt eine Variable.

Missing left paranthesis

Bei *GENCOM* müssen bei der Definition des SCB (System Control Block) die Parameter in Klammern stehen. Die linke Klammer fehlt.

Missing right paranthesis

Bei *GENCOM* müssen bei der Definition des SCB (System Control Block) die Parameter in Klammern stehen. Die rechte Klammer fehlt.

Missing SCB value.

Bei der SCB-Definition von *GENCOM* fehlen die Werte.

More than four drives specified.

Bei der SETDEF-Suchpfad-Definition wurden mehr als vier Laufwerke angegeben.

MULTIPLE DEFINITION:

Es wurde ein Symbol (Anzahl der signifikanten Zeichen ist wichtig!) mehrfach definiert. Diesen Fehler meldet der Linker.

n?

Sie haben bei der Benutzerbereichsdefinition einen zu großen Wert angegeben (0-15 ist erlaubt).

No directory label exists.

Die LABEL-Option wurde bei SHOW verwandt, obwohl die Diskette kein Label hat.

No directory space
NO DIRECTORY SPACE - <Dateiname>

Nicht genügend Platz im Inhaltsverzeichnis (maximal 64 oder 128 Einträge) für *COPYSYS*, *GENCOM*, *MAC*, *PIP*, *RMAC* oder *SAVE*. Löschen Sie unnötige Dateien.

No disk space.

SAVE benötigt mehr Platz auf der Diskette als vorhanden. Löschen Sie unnötige Dateien, oder benutzen Sie eine andere Diskette.

No file
NO FILE:
NO FILE - <Dateiname>

Der eingegebene Dateiname kann bei *DIR*, *ERASE*, *LIB*, *LINK*, *PATCH*, *PIP*, *RENAME* oder *TYPE* auf dem definierten Laufwerk nicht gefunden werden.

No HELP.HLP file on the default drive.

HELP.HLP muß auf dem Standardlaufwerk sein.

NO INPUT FILE PRESENT ON DISK

Der bei *DUMP* angegebene Dateiname existiert nicht.

No memory

Es ist nicht mehr genügend Speicher vorhanden, um das gewählte Programm (COM-Datei) zu laden.

No modifier for this option.

Unnötige Angabe von einer Option bei Aufruf von *GENCOM*.

NO MODULE:

Das bei *LIB* angegebene Modul kann nicht gefunden werden.

No more space in the header for RSX or SCB initialization

Der (Datei-)Header hat lediglich Platz für 15 Einträge; die Anzahl an Kombinationen von RSX und SCB übersteigt die Grenze.

No options specified.

Bei *SET* wurden keine Optionen angegeben.

No PRN file.

XREF kann die Datei {Dateiname.PRN} nicht finden.

No Records Exist

DUMP findet lediglich den Directoryeintrag einer Datei, aber keine entsprechenden Records.

No source file on disk.

COPYSYS kann die Systemdatei CPM3.SYS auf der definierten Diskette nicht finden.

NO SOURCE FILE PRESENT:

Die bei *MAC* oder *RMAC* angegebene Quelldatei existiert nicht.

NO SPACE

SAVE meldet, daß keine Einträge mehr im Inhaltsverzeichnis möglich sind.

No 'SUB' file found.

Die ausgewählte SUB-Datei kann nicht gefunden werden.

No such file to rename.

Die umzubenennende Datei kann nicht gefunden werden.

No SYM file

XREF kann keine Datei <Dateiname.SYM> finden.

NON-SYSTEM FILE(S) EXIST

Es wurde *DIRS(YS)* verlangt, aber keine Systemdatei gefunden.

Not enough available memory.
Not Enough Memory
Nor Enough Memory for Sort.

Nicht genügend Speicher vorhanden für Sortieroperation.

Not enough room in directory.

Für die Zeitstempelfunktion ist nach Eingabe von *INITDIR* leider nicht mehr genügend Platz auf der Diskette.

NOT FOUND

PIP kann die gewünschte Datei nicht finden.

Not renamed, <Datei> read only.

Die angegebene Datei kann nicht umbenannt werden, da sie Read-Only ist.

OPEN FILE NONRECOVERABLE

Die bei *SET* angegebene Option ist nicht für Dateien, sondern lediglich für Disketten erlaubt.

Option requires a file reference.

Die angegebene Option benötigt einen Dateinamen.

Out of data space.

COPYSYS kopiert die Systemdatei CPM3.SYS; während des Kopierens ist die Diskette voll.

Options not grouped together

Die Optionen müssen alle innerhalb einem eckigen Klammerpaar angegeben werden.

Output File Exists, Erase it.

Die angegebene Ausgabedatei darf nicht bereits existieren.

OUTPUT FILE READ ERROR:

MAC oder *RMAC* können keine Ausgabedatei anlegen, weil die Diskette voll ist. Löschen Sie unnötige Dateien.

OVERLAPPING SEGMENTS:

Der Linker erkennt einen Versuch, ein Speichersegment in den Speicher zu schreiben, der bereits von einem anderen Segment belegt wird.

Page and no page option selected.
No page in effect

Sie haben beide Optionen angegeben. Nur die NO PAGE-Option wird berücksichtigt.

Parameter Error

In SUBMIT-Dateien können die Parameter lediglich mit $0 bis $9 durchnumeriert werden.

Password Error.

Bei *DUMP*, *ERASE*, *GENCOM* oder *TYPE* ist ein falsches Paßwort eingegeben worden.

Physical Device Does Not Exist.

Das bei *DEVICE* angegebene Gerät existiert nicht.

Possible incompatible disk format.

Quell- und Zieldiskette haben bei *COPYSYS* verschiedene Formate.

PROGRAM INPUT IGNORED.

Dieser Meldung wird 'Warnung' vorangestellt; in einer SUBMIT-Datei sollen Werte beispielsweise an *PIP* durch < übergeben werden. Das Programm benötigt allerdings keine weiteren Eingaben. Die SUBMIT-Datei wird fortgeführt. Siehe *PIP* und *SUBMIT*.

PUT>

Dieses Prompt kommt, wenn bei *PUT* die ECHO-Option eingeschaltet worden ist.

PUT ERROR: FILE ERASED.

Die PUT-Ausgabedatei wurde gelöscht, kann allerdings nicht geschlossen werden.

QUIT NOT FOUND

Der im PIP-Kommando angegebene String mit Q-Option konnte in der Datei nicht gefunden werden.

Random Read

Während dem Einlesen der temporären Datei einer SUBMIT-Datei ist ein Fehler aufgetreten.

Read only.

Das Laufwerk der angegebenen Datei bei *GENCOM* oder *SET* ist schreibgeschützt.

Read error

Ein Fehler beim Lesen der bei *TYPE* angegebenen Datei. Überprüfen Sie die Diskette.

Reading file: <Dateiname>

Ein Fehler ist aufgetreten beim Versuch, die Datei <Dateiname> zu lesen.

Reading file HELP.HLP
Reading HELP.HLP index

HELP-Datei wurde nicht gefunden, oder es ist ein Fehler während des Lesevorganges aufgetreten. Überprüfen Sie die Diskette oder kopieren Sie die HELP-Datei von der Originaldiskette.

RECORD TOO LONG

Bei *PIP* mit H-Option sind mehr als 80 Zeichen eingelesen worden, somit ist der Buffer voll.

Requires CPM 3.0 or higher.

Eines der Hilfsprogramme *DATE*, *DEVICE*, *DIR*, *ERASE*, *GENCOM*, *HELP*, *INITDIR*, *PIP*, *SET*, *SETDEF*, *SHOW*, *RENAME* oder *TYPE* wurde nicht unter CP/M 3.0 gestartet.

R/O FILE

Die Zieldatei bei *PIP* ist Read-Only und kann nicht überschrieben werden.

Sort Stack Overflow

Beim Sortiervorgang von *DIR* mit Sortieroption reicht der verfügbare Speicher nicht aus, um die Funktion ordnungsgemäß zu Ende zu führen.

SOURCE FILENAME ERROR:

Die Syntax des eingegebenen Quelldateinamens für *MAC* oder *RMAC* ist nicht korrekt.

START NOT FOUND

Der eingegebene String-Ausdruck bei *PIP* in Verbindung mit der S-Option konnte in der angegebenen Datei nicht aufgefunden werden.

Symbol Table reference overflow
Symbol Table overflow

XREF meldet, daß nicht genügend Speicher für die Symboltabelle vorhanden ist.

Too many entries in Index Table.
Not enough memory

Nicht genügend Speicher vorhanden, um die Stichworttabelle des HELP-Kommandos einzuladen.

Topic: <String> Not found.

Das gewünschte Stichwort konnte in der Datei HELP.HLP nicht gefunden werden. *HELP* zeigt Ihnen alle erlaubten Stichworte an

Total file size exceeds 64K.

Eine Ausgabedatei ist größer als 64 KByte. Dies ist bei Ihrem Commodore oder Ihrem Schneider normalerweise möglich, nicht aber unter CP/M, da im Inhaltsverzeichnis nicht mehr Platz vorgesehen ist.

Try 'PAGE' or 'NO PAGE'

Bei *TYPE* wurde eine andere als die beiden oben genannten Optionen angegeben.

Unable to allocate Data deblocking buffer space.

Nicht mehr genügend Platz im Systemspeicher vorhanden, um eine Datenentblockung vorzunehmen.

Unable to allocate space for hash table.

Nicht mehr genügend Speicher vorhanden, um Sortierfunktion ausführen zu können (Anlegen einer Hash-Tabelle).

Unable to close HELP.DAT.
Unable to close HELP.HLP.

Beim Schließen einer der beiden Dateien HELP.HLP oder HELP.DAT ist ein Fehler aufgetreten. Wahrscheinlich ist nicht mehr ausreichend Speicherplatz auf der Diskette frei oder das Inhaltsverzeichnis bereits voll.

Unable to find HELP.HLP.

Um Ihnen helfen zu können, benötigt *HELP* die Datei HELP.HLP. Diese kann nicht gefunden werden. Kopieren Sie sie nötigenfalls von Ihrer Systemdiskette.

Unable to Make HELP.DAT.
Unable to Make HELP.HLP.

Die Dateien sind entweder schreibgeschützt, oder es ist. nicht mehr genügend Speicherplatz auf der Diskette

UNBALANCED MACRO LIBRARY.

Eine begonnene Makrodefinition bei *MAC* oder *RMAC* wurde nicht vor EOF-Marke geschlossen.

UNDEFINED START SYMBOL:

Der Linker hat ein Symbol gefunden, das in keinem der gelinkten Module definiert worden ist.

UNDEFINED SYMBOLS:

Der Linker zählt nun alle Symbole auf, die zwar irgendwo angesprochen wurden, aber in keiner der gelinkten Dateien definiert worden sind.

UNEXPECTED END OF HEX FILE - <Dateiname>

PIP ist auf ein unerwartetes EOF gestoßen. Ein Hex-Record ist nicht korrekt abgeschlossen.

Unrecognized drive.

Es wurde bei *SHOW* ein Laufwerk angegeben, das nicht erlaubt ist. (Erlaubt: A - P)

Unrecognized input.

Das eingegebene SHOW-Kommando hat nicht das vorgeschriebene Format.

Unrecognized option

Bei *GENCOM* oder *SHOW* wurde eine unbekannte, unerlaubte oder falsche Option eingegeben.

USER ABORTED

PIP wurde unterbrochen, indem Sie CTRL-C betätigt haben.

VERIFY ERROR: - <Dateiname>

Sie haben *PIP* durch Angabe der V-Option zur Korrektur beim Kopieren aufgefordert. *PIP* hat eine Differenz gefunden und meldet diese.

<Dateiname> Write error

XREF teilt Ihnen mit, daß nicht mehr genügend Speicher auf der Diskette frei ist, um die angegebene Datei zu speichern.

Write protected?

Die bei *COPYSYS* vorgesehene Zieldiskette ist schreibgeschützt oder Read-Only.

Writing file: <Dateiname>

Beim Schreiben der angegebenen Datei ist bei *HELP* ein Fehler aufgetreten.

Wrong Password.

Das bei *SET* eingegebene Paßwort ist falsch oder ungültig.

CTRL-	Ausführung
A	*Cursor ein Zeichen nach links*
B	*Cursor an den Anfang der Zeile bzw. an das Zeilenende, sollte der Cursor bereits am Zeilenanfang stehen*
C	*CP/M-Funktion abbrechen bzw. CP/M neu booten*
E	*Cursor in die nächste Zeile*
F	*Cursor ein Zeichen nach rechts*
G	*Zeichen unter Cursor löschen*
H	*Zeichen links von Cursor löschen*
I	*Cursor an die nächste Tabulatorposition*
K	*Lösche von Cursor bis Zeilenende*
M	*Entspricht der Taste <RETURN>*
P	*Druckerprotokoll ein- bzw. ausschalten*
Q	*Weiterdrucken nach ^S (XON/XOFF)*
R	*Befehlszeile wiederholen*
S	*Ausgabe anhalten, mit ^Q weiterdrucken (XON/XOFF)*
U	*Gesamte Zeile löschen*
W	*Alte Befehlszeile wieder anzeigen*
X	*Zeichen links von Cursor löschen*
Z	*Ende Zeichenkette bei PIP oder ED etc.*

BDOS und BIOS

BDOS und BIOS

Programme werden unter CP/M erst dadurch kompatibel, daß alle Programme fest definierte Unterprogramme (beispielsweise "Zeichen ausgeben") aufrufen, die die hardwarespezifischen Eigenschaften des jeweiligen Rechners berücksichtigen. Diese Unterroutinen hat man in zwei logische Einheiten unterteilt:

BIOS = *B*asic *I*nput *O*utput *S*ystem
BDOS = *B*asic *D*isc *O*peration *S*ystem

Auf diese Weise kann es sich der Programmierer nun sparen, für jeden Rechner seine Programme umzuschreiben. Hat er oder jemand anders einmal die Routinen des *BIOS* und des *BDOS* umgeschrieben, so laufen alle CP/M-Programme auf dem neuen Rechner.

Von der ersten CP/M-Version bis zur heutigen aktuellen CP/M 3.0 hat sich einiges getan. Einige Kommandos sind verschwunden, andere hinzugekommen oder ergänzt worden. Auch *BIOS* und *BDOS* hat man ergänzt, einmal bestehende und definierte Routinen jedoch nicht mehr geändert, höchstens erweitert, aber so, daß man unter den alten Voraussetzungen diese weiter benutzen kann. Zur Erhaltung eines Standard eine Grundbedingung.

Die *BIOS*-Routinen sind die geräteabhängigeren Routinen, sie werden auf einer niedrigeren Ebene durchgeführt als *BDOS*-Routinen. Man kann aber praktisch auf den Aufruf dieser Routinen verzichten, weil es zumeist äquivalente *BDOS*-Routinen gibt. Ferner können Sie durch die *BDOS*-Routine 50 beliebig *BIOS*-Routinen aufrufen. Bedenken Sie auch, daß *BIOS* und *BDOS* verschiebbar sind. Die *BDOS*-Routinen rufen Sie *alle* an Adresse &0005H auf, die Einsprungadressen der *BIOS*-Routinen errechnen Sie aus:

*<Tabellenstart> + 3 * BDOS-Referenznummer*

was viel schwieriger und fehleranfälliger ist. Bedenkt man dann noch, daß man bei CP/M-Plus-Rechnern auch das Banking

berücksichtigen muß, so ergibt sich die Benutzung von *BDOS*-Routinen fast zwingend.

Will man eine der vorhandenen *BDOS*-Routinen aufrufen, so muß man die Referenznummer der Routine (beginnend mit 0) im C-Register übergeben und dann den *BDOS*-Routinentreiber an Adresse &0005H aufrufen. Dies sieht im Maschinenprogramm etwa so aus:

```
bdos    equ   0005h          ;BDOS Einsprungadresse
conin   equ   1              ;BDOS Eingaberoutine Tastatur

        ;
        ;Hauptprogramm
        ;*************
        ;

        org   0100h          ;Beginn aller Programme
next:   mvi   c, conin       ;Tastatureingabe
        call  bdos           ;BDOS-Routine ausführen
        cpi   13             ;<CR> ist Ende
        jnz   next           ;Nein, dann weiter
        ret
        end
```

BIOS-Routinen

N°	*Name*	*Eingabe*	*Ausgabe*
0	BOOT	Keine	Keine
1	WBOOT	Keine	Keine
2	CONST	Keine	A:0FFH fertig A:00H nicht fertig
3	CONIN	Keine	A:Zeichen
4	CONOUT	C:Ausgabezeichen	Keine
5	LIST	C:Zeichen	Keine
6	AUXOUT	C:Zeichen	Keine
7	AUXIN	Keine	A:Zeichen
8	HOME	Keine	Keine

9	SELDSK	C:Laufwerk (0-15)	HL:DPH-Adresse
		E:Init Sel Flag	HL:0000H ungültig
10	SETTRK	BC:Tracknr.	Keine
11	SETSEC	BC:Sektornr.	Keine
12	SETDMA	BC:DMA	Keine
13	READ	Keine	A:00H kein Fehler
			A:01H Fehler
			A:0FFH Diskettenwechsel
14	WRITE	C:Deblk Code	A:00H kein Fehler
			A:01H Fehler
			A:02H Diskette ist $RO
			A:0FFH Diskette gewechselt
15	LISTST	Keine	A:00H nicht fertig
			A:0FFH fertig
16	SECTRN	BC:Sektornr.	HL:Physikal. Sektornr.
		DE:Trans Tbl Adr	
17	CONOST	Keine	A:0FFH fertig
18	AUXIST	Keine	A:00H nicht fertig
			A:0FFH fertig
19	AUXOST	Keine	A:00H nicht fertig
			A:0FFH fertig
20	DEVTBL	Keine	HL:Zeichentabelle Adresse
21	DEVINI	C:Gerät (0-15)	Keine
22	DRVTBL	Keine	HL:Laufwerkstabelle Adresse
			HL:0FFFEH oder HL:0FFFFH
23	MULTIO	C:Sektorzähler	Keine
24	FLUSH	Keine	A:00H kein Fehler
25	MOVE	HL:Zieladresse	HL und DE zeigen auf
		DE:Quelladresse	Folgebytes
		BC:Anzahl	
26	TIME	C=Setzen/Holen	Keine
27	SELMEM	A:Speicherbank	Keine
28	SETBNK	A:Speicherbank	Keine
29	XMOVE	B:Zielbank	Keine
		C:Quellbank	
30	USERF	Reserviert	
31	RESERV1	Reserviert	
32	RESERV2	Reserviert	

BDOS-Routinen:

N°	*Beschreibung*	*Eingabe*	*Ausgabe*
0	System Reset	Keine	Keine
1	Konsoleingabe	Keine	A:Zeichen
2	Konsolausgabe	E:Zeichen	A:00H
3	Lochstreifen ein	Keine	A:Zeichen
4	Lochstreifen aus	E:Zeichen	A:00H
5	Druckerausgabe	E:Zeichen	A:00H
6	Direkte Konsolen	E:0FFH	A:Zeichen/Status
	Ein-/Ausgabe	0FEH	Keine
		0FDH oder Zeichen	
7	IOBYTE abfragen	Keine	A:00/0FFH
8	IOBYTE setzen	E:00/0FFH	Keine
9	Ausgabe String	DE:.String	A:00H
10	Eingabe Buffer	DE.Buffer	A:Zeichen
11	Konsolenstatus	Keine	A:00/01
12	CP/M-Version	Keine	HL:Version (0031H)
13	Reset Disk System	Keine	A:00H
14	Diskette ausw.	E:Laufwerk	A:Fehlerflag
15	Öffnen Datei	DE:.FCB	A:Fehler
16	Schließen Datei	DE:.FCB	A:Fehler
17	Erster Eintrag	DE:.FCB	A:Fehler
18	Nächster Eintrag	DE:.FCB	A:Fehler
19	Datei löschen	DE:.FCB	A:Fehler
20	Sequentiell lesen	DE:.SCB	A:Fehler
21	Sequentiell schreiben	DE:.SCB	A:Fehler
22	Datei einrichten	DE:.FCB	A:Fehler
23	Datei umbenennen	DE:.FCB	A:Fehler
24	Loginvektor ermitteln	Keine	HL:Loginvektor
25	Standardlaufwerk	Keine	A:Standardlaufwerk
26	DMA-Adresse definieren	DE:.DMA	A:00H
27	Tabelle ermitteln	Keine	HL:.Zugriffsadresse
28	Laufwerk schützen	Keine	A:00H
29	$RO-Laufwerke	Keine	HL:.$RO-Vektor
30	Dateiattribute setzen	DE:.FCB	A:Fehler
31	Diskettenparameter	Keine	HL:.DPB
32	Benutzernr.	E:0FF/	A:Aktueller User
	setzen/holen	Usernummer	00H

33	Random lesen	DE:.FCB	A:Fehler
34	Random schreiben	DE:.FCB	A:Fehler
35	Dateigröße	DE:.FCB	A:Fehlerflag (sowie r0, r1, r2)
36	Random Record setzen	DE:.FCB	r0, r1, r2
37	Laufwerk Reset	DE:Laufwerk	A:00H
38	Zugriff Laufwerk	Keine	A:00H
39	Laufwerk befreien	Keine	A:00H
40	Random schreiben mit Löschen	DE:.FCB	A:Fehler
41	Record testen/schreiben	DE:.FCB	A:0FFH
42	Record schützen	DE:.FCB	A:00H
43	Record freigeben	DE:.FCB	A:00H
44	Multi-Sektor-Zähler	E:# Sektoren	A:Rückgabewert
45	BDOS-Fehler-Modus ein	E:Modus	A:00H
46	Freier Diskettenspeicher	E:Laufwerk	Anzahl freier Sektoren A:Fehler
47	Programm verketten	E:Chain Flag	A:00H
48	Buffers räumen (flush)	E:Purge Flag	A:Fehler
49	Holen/Setzen System Control Block	DE:.DCB	A:Rückgabe-Byte HL:Rückgabe-Wort
50	Direkte BIOS-Aufrüfe	DE:.BIOS	BIOS-Rückgabe
59	Load Overlay	DE:.FCB	A:Fehler
60	Aufruf RSX	DE:.RSX	A:Fehler
98	Blöcke freigeben	Keine	A:Fehler
99	Datei abschneiden	DE:.FCB	A:Fehler
100	Directory Label setzen	DE:.FCB	A:Fehler
101	Directory Label Informationen holen	E:Laufwerk	A:Dir Label Data Byte
102	Paßwortmodus und Datumsstempel holen	DE:.FCB	A:Dir Code
103	Datei schreiben XFCB	DE:.FCB	A:Dir Code
104	Datum und Zeit setzen	DE:.DAT	A:00H
105	Datum und Zeit holen	DE:.DAT	Datum und Zeit; A:Sekunden
106	Default-Paßwort setzen	DE:.Paßwort	A:00H
107	Seriennummer holen	DE:.#-Feld	Seriennummer
108	Hole/Setze Programm Rückgabecode	DE:0FFFFH Modus	HL:Programmrückgabecode Keine
109	Hole/Setze Konsolmodus	DE:0FFFFH Modus	A:Konsolenmodus Keine

110	Hole/Setze Ausgabe Begrenzer	DE:0FFFFH E:Begrenzer	A:Ausgabebegrenzer Keine
111	Block drucken	DE:.CCB	A:00H
112	Block drucken (Drucker)	DE:.CCB	A:00H
152	Dateinamen überprüfen	DE:.PFCB	

DPB: Disk Parameter Block
DPH: Disk Parameter Header
FCB: File Control Block
SCB: System Control Block
RSX: Resident System Extension
BCB: Buffer Control Block

Führender Punkt (.FCB): Zeiger auf (Adresse)

Handhabung

Das Promptsymbol

Haben Sie CP/M auf Ihrem Rechner gebootet, so erscheint das sogenannte *Promptsymbol* auf Ihrem Bildschirm. In diesem Promptsymbol versteckt sich das aktuelle Standardlaufwerk sowie bei CP/M 3.0 der aktuelle Benutzerbereich, sofern dieser nicht gleich Null ist.

CP/M 2.2	*CP/M 3.0*
A>	*3A>*

Residente und transiente Kommandos

Sie können nun Kommandos über die Tastatur eingeben. Dabei können diese Kommandos *residente* Kommandos sein, auf die Sie jederzeit zurückgreifen können, da sich diese Programme im Speicher Ihres Rechners befinden. Die *transienten* Kommandos hingegen müssen sich auf der im Standardlaufwerk eingelegten Diskette befinden. Transiente Kommandos erkennen Sie daran, daß sie die Dateikennung .COM tragen. Mit folgendem Befehl können Sie sich alle transienten Kommandos der aktuellen Diskette ausgeben lassen:

```
DIR *.COM
```

Wollen Sie eines der in diesem Führer beschriebenen transienten Kommandos verwenden, so muß sich dies auch in der ausgegebenen Liste befinden.

Submit-Dateien

Neben den transienten Kommandos können Sie auch die sogenannten *Batch*-Dateien oder *Submit*-Dateien durch einfaches Eingeben des Dateinamen ohne Dateikennung aufrufen. Diese Dateien tragen die Endung .SUB.

Sollte eine Datei sowohl mit der Endung .SUB als auch mit der Endung .COM existieren, so wird die Datei gestartet, die die erste mittels SETDEF definierte Kennung trägt.

Der Suchpfad

Ebenfalls mittels SETDEF können Sie definieren, in welcher Reihenfolge die vorhandenen Laufwerke zum Auffinden des eingegebenen Dateinamen herangezogen werden. Normalerweise wird zuerst auf dem Standardlaufwerk gesucht, es sei denn, die Laufwerksbezeichnung wurde explizit angegeben. Beachten Sie dabei, daß grundsätzlich die Laufwerksbezeichnungen A bis P möglich sind.

Beachten Sie bei CP/M 3.0 auch, daß zunächst unter dem aktuellen Benutzerbereich die Kommandodatei gesucht wird. Ist diese Suche hier erfolglos, wird unter Benutzerbereich Null ebenfalls gesucht. Deswegen hat dieser Benutzerbereich auch eine Sonderstellung.

Kann CP/M das eingegebene Kommando nicht im aktuellen und nicht im Benutzerbereich Null finden, und ist auch die weitere Suche auf den in Folge angegebenen Laufwerken erfolglos, dann gibt CP/M den Dateinamen in Großbuchstaben mit einem Fragezeichen versehen aus.

```
RUN?
```

Wichtiger Tip: Die COM-Dateien im Benutzerbereich Null haben allerdings nur dann die erwähnte Sonderstellung, hat man sie mit dem SET-Kommando und der SYS-Option versehen!

Mehrere Kommandos in einer Zeile

Auch ist es möglich, mehrere Kommandos in einer Zeile einzugeben. Dazu muß man die verschiedenen Kommandos natürlich voneinander trennen. Hierzu verwendet man das Trennsymbol »!«. Gerade bei Submit-Dateien kann dies sehr sinnvoll sein, weil der *CCP* (Console Command Processor) für jede Eingabezeile von Diskette lädt, was relativ zeitaufwendig ist. Je mehr Kommandos in einer Zeile, umso weniger muß der CCP nachladen.

Ein Beispiel für eine Eingabezeile mit dem Trennsymbol ! wäre die folgende Kommandoaufreihung:

```
DIR *.COM!DIRS!SHOW [LABEL]
```

Automatisches Starten von Dateien

Will man nach dem Booten des Systems eine spezielle Befehlsfolge immer wieder automatisch ablaufen lassen, so kann man diese Befehlsfolge zusammenfassen und dann mittels dem Submit-Kommando relativ einfach abrufen. Noch einfacher ist es allerdings, wenn Sie diese Kommandos in eine spezielle Submit-Datei mit Namen PROFILE.SUB (bei manchen CP/M-Versionen AUTORUN.SUB o.ä.) schreiben. Diese Datei PROFILE.SUB wird nach dem Booten automatisch abgearbeitet.

Programm abbrechen

Die meisten CP/M-Programme - nicht aber zwingendermaßen alle - können durch die Tastenkombination CTRL-C unter-

brochen werden. Drücken Sie die CTRL-Taste und halten Sie sie fest; dann betätigen Sie die C-Taste. Auf dem Bildschirm erscheint zumeist *^C*. Diese Taste führt aber auch ein RESET beim Diskettenlaufwerk aus; man nennt dies auch einen *Warmstart*. Zusätzlich definierte Laufwerke neben dem Standardlaufwerk sind dann nicht mehr bekannt und müssen erneut mittels dem SETDEF-Kommando definiert werden.

Hilfe

Eine Hilfe können Sie von CP/M jederzeit erwarten - vorausgesetzt, Sie haben die entsprechende Systemdiskette im Standardlaufwerk. Entweder geben Sie HELP direkt mit einem Stichwort ein - Sie erhalten dann die vorhandenen Informationen über dieses Stichwort - oder Sie geben HELP ohne Stichwörter ein; dann erhalten Sie eine Liste aller möglichen Stichworte, woraus Sie sich eines auswählen können.

Literaturverzeichnis

Das CP/M-Buch zum Commodore 128
Schieb, Weiler
1985 Data Becker Verlag, Düsseldorf
ISBN 3-89011-116-5

User's Guide / Programmer's Guide / System Guide
CP/M Version 3 - Operating System
Digital Research
1983 Digital Research Inc.
319791-01

Das CP/M-Trainingsbuch zum CPC
Schieb, Weiler
1985 Data Becker Verlag, Düsseldorf
ISBN 3-89011-089-4

Das große Floppybuch zum CPC
Brückmann, Schieb
1986 Data Becker Verlag, Düsseldorf
ISBN 3-89011-093-2

A

AMSDOS *14*
Archiv-Flag *62ff*
Assembler *22, 45, 50, 62*
ASM *22*
Attribute *62ff, 79ff*
Ausgabe *58f*
Ausgabe Datei *19, 29, 53ff, 84f*
AUXIN *79ff, 26f*
Ausgabekanal *26f*
AUXOUT: *79ff, 26f*

B

Batch *77f*
Baudrate *27*
BDOS *119*
Befehlsübersicht *9*
Benutzerbereich *73, 20, 79ff*
Betriebssystem *11*
Bibliothek *45, 46ff*
Bildschirmausgabe umleiten *58f*
BIOS *119*

C

CCP *88*
CONIN: *79ff, 26f*
CONOUT: *79ff, 26f*
Console *38f, 58f, 79ff*
Control-Codes *118*

COPYSYS *23*
CP/M-Version erstellen *37*
CP/M-Versionen *11*
CPU *11*
Crossreference *85*

D

DATE *24*
Datei ausgeben *19, 29, 53ff, 84f*
Datei löschen *35*
Datei umbennen *60f*
Datei anzeigen *73*
Datei drucken *19, 29, 53ff, 84f*
Datei kopieren *53ff*
Dateiattribut *62ff, 79ff*
Dateiinhalt ausgeben *19, 84f*
Datumsabfrage *24*
Datumsstempel *43*
DDT *25*
Debugger *25, 74*
DEVICE *26*
DIR *15, 28f*
Diskette formatieren *36*
DIRSYS *16*
DISCKIT3 ..... *36*
Disketteninhalt *15, 28f*
Diskettenlabel *73*
Drucken *53ff*
Drucker *26f, 58f*
DUMP *29*

E

ED *30ff*
Editor *30ff*
Eingabedatei *38f*
Eingabekanal *26f, 38f*
Einsprungadresse *120*
ERASE *17, 35*
Error *87*
Erstellen einer COM-Datei *49*

F

Fehlermeldungen *87*
FORMAT *35*

G

GENCOM *37*
GET *38f*

H

HELP *39f*
Hex-Format .....
HEXCOM *42*
Hilfe *39f*

I

INITDIR *43*
Inhaltsverzeichnis *43*
Inhaltsverzeichnis anz *15, 28f, 73*

J

Joker *28f*

K

Kanal *26f*
KEYFIG *44*
Konsole *38f, 58f, 79ff*
Kontrollzeichen *118*
Kopieren von Dateien *53ff*
Kopieren des Systems *23, 83*

L

Löschen Datei *17, 35*
LIB *45*
LINK *46*
LOAD *49*
LST: *26f*

M

Makroassembler *50, 62*
Maschinensprache *22, 25, 42, 45, 46ff, 49*
Maschinensprache *50, 62, 63f, 74ff, 85*
MAC *50*
Mnemocode *22, 50*
Monitorprogramm *25, 74ff*
MOVCPM *51*

P

PATCH *52*
Paßwort *62ff, 73*
Peripheriegeräte *26f*
PIP *53ff*
Protect *62ff*
Prozessor *11*
PUT *58f*

R

RENAME *18, 60f*
Residente Kommandos *12, 13*
RSX *37*
Rückkehr zum AMSDOS *14*

S

SAVE *63f*
SET *62ff*
SETDEF *69f*
SETUP *71f*
SHOW *73*
SID *74ff*
Speicherinhalt sichern *63f*
Speicherplatz, freier *73*
Standard *11*
Standardlaufwerk definieren *69f*
STAT *79f*
Stapelverarbeitung *77f*
Startkonfiguration definieren *71f*
Suchpfad *69f*
SUBMIT *77f*
Symboltabelle *85*
System kopieren *23, 83*
System verschieben *51*
SYSGEN *83*

Systemdateien *16, 28f*
Systemdateien anzeigen *16, 28f*

T

Tastatur-Kontrollzeichen *118*
Tastaturbelegung *44*
TPA *83*
Transiente Kommandos *12, 21*
TYPE *19, 84f*

U

Umbenennen *18, 60f*
Userbereich *73, 79ff*
USER *20*

V

Vektortabelle *120*
Verschieben des Systems *51*
Version erstellen *37*

W

Warnungen *87*
Wild Cards *28f*

X

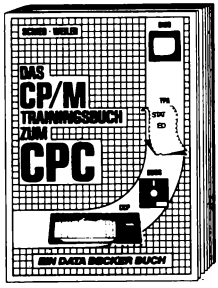
XON/XOFF *27*
XREF *85*

Z

Zeileneditor *30ff*
Zeitabfrage *24*
Zeitstempel *43, 62ff*
Zusammenführen *46ff*

Für alle, die mehr wissen wollen...

... über CP/M auf ihrem Rechner gibt es die großen CP/M-Bücher zum Commodore 128 und Schneider CPC. Denn wer CP/M wirklich beherrschen will, der muß mehr wissen als nur die Systemkommandos: Grundlagen und Möglichkeiten des CP/M-Betriebssystems, CP/M intern mit Speicherverteilung und ROM-Listing, Systemprogrammierung mit den Makro-Assemblern MAC und RMAC, SUBMIT-Dateien, fremde Diskettenformate und Sonderfunktionen.

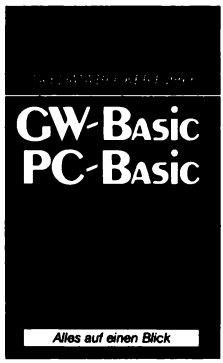


Schieb/Weiler
Das CP/M-Buch zum Commodore 128
340 Seiten, ISBN 3-89011-116-5, DM 49,-

Schieb/Weiler
Das CP/M-Trainingsbuch zum CPC
260 Seiten, ISBN 3-89011-089-4, DM 49,-

In der Regel klappt sie phantastisch, die Arbeit mit dem Computer. Und für Zweifelsfälle hat man ja bereits eine ansehnliche Bibliothek nützlicher Literatur. Doch immer wieder – mitten in der Arbeit – passiert es: Man sucht nach einem bestimmten Kommando. Irgendwo im Handbuch, oder stand es in einem Computermagazin... Der Arbeitsfluß ist unterbrochen. Man versucht sich zu erinnern, durchwühlt den riesigen Literaturberg, sucht einen Hinweis. HILFE. Genau die bekommen Sie von den neuen DATA BECKER Führern. Ein gezielter Griff und Sie haben die gewünschte Information. Hier finden sie umfassend alles auf einem Blick. Zu Ihrem Rechner oder auch zur entsprechenden Software. Das sind die ersten DATA BECKER Führer:

Der DATA BECKER Führer zu TURBO PASCAL
128 Seiten
ersch. ca. Anfang Nov.
DM 24,80



Der DATA BECKER Führer zu GW-BASIC/PC-BASIC
160 Seiten
ersch. ca. Anfang Nov.
DM 24,80

Wie gesagt, dies ist erst der Anfang. Weitere DATA BECKER Führer werden folgen: First Word/First Lektor, C16, CPC, WordStar, MS-DOS und, und, und. Immer aktuell, übersichtlich und hilfreich. Immer im gleichen Gewand: Robustes Hardcover im handlichen Westentaschenformat. Und immer gleich strukturiert: Alle Befehle und Funktionen nach Sachgruppen, alphabetisch mit Kurzsyntax und über Stichworte geordnet. Egal, wie sich Ihr Problem darstellt, mit einem Blick in den DATA BECKER Führer ist es bereits gelöst.

ISBN 3-89011-400-8

DM 19,80
SFr 19
ÖS 154

This was brought to you

from the archives of

http://retro-commodore.eu